Tampereen yliopisto
Tampere University

TIE-02306

Introduction to Software Engineering

5 credit units

12-emb-ItSE-2019-v3

Tampereen yliopisto
Tampere University

Current at course (w 47)

- **WE10 next week**
- **continue updating your Trello (kanban) boards** = use at your process

- **2nd presentations were,**
- **group-to-group feedback at PRP open up to 01.12.2019**
- **final versions of documentation 08.12.2019**
- **peer feedback inside own group, and self assesment**

- **EXAM 3/3 at weeks 47-49; lectures 7-10, 12 questions (right/wrong) in 6 sections + volunteer feedback.**

  (first tryout: 38 min, 12/24 points)

Backlog items with deadline

- **09.09.2019** at 23:59 Group forming (Moodle)
- **15.09.2019** at 23:59 Trello creation (Trello)
- **13.10.2019** at 23:59 Phase 1 documentation (Moodle)
- **13.10.2019** at 23:59 Phase 1 presentation slides (PRP-tool)
- **Week 43** Phase 1 presentations (Physical realm)
- **03.11.2019** at 23:59 Phase 1 peer feedback (PRP-tool)
- **17.11.2019** at 23:59 Phase 2 documentation (Moodle)
- **17.11.2019** at 23:59 Phase 2 presentation slides (PRP-tool)
- **Week 47** Phase 2 presentation (Physical realm)
- **01.12.2019** at 23:59 Phase 2 peer feedback (PRP-tool)
- **08.12.2019** at 23:59 Final delivery of project documentation (Moodle)
- **15.12.2019** at 23:59 Final peer feedback and self assessments (PRP-tool).

Tampereen yliopisto
Tampere University

Weekly exercise attendees

	w36 WE1	w37 WE2	w38 WE3	w39 WE4	w40 WE5	w41 WE6	w44 WE7	w45 WE8	w46 WE9	w48 WE10
WED	0	14	9	5	8	9	9	7	7	
THU	21	13	14	17	16	13	6	17	12	

We will continue two Weekly Exercise groups, as long as the number of attendees are reasonable.

Tampereen yliopisto
Tampere University

Embedded systems

Embedded systems

Embedded systems (FI: sulautetut järjestelmät)
have software and hardware working closely together.

For example

- mobile phones
- lift/elevator
- coffee maker and other kitchen machines
- DVD player
- television
- aircraft

Embedded software

- Computers are used to **control** a wide range of systems from simple domestic machines, through games controllers, to entire manufacturing plants.
- Their software must **react to events** generated by the hardware and, often, issue control signals in response to these events.
- The **software in these systems is embedded in system hardware**, often in read-only memory, and usually **responds**, in real time, to events from the system's environment.

Responsiveness

- Responsiveness in real-time is the critical difference between embedded systems and other software systems, such as information systems, web-based systems or personal software systems.
- For non-real-time systems, correctness can be defined by specifying how system inputs map to corresponding outputs that should be produced by the system.
- In a real-time system, the correctness depends both on the response to an input and the time taken to generate that response. If the system takes too long to respond, then the required response may be ineffective.

Definitions

- A real-time system is a software system where the correct functioning of the system depends on the results produced by the system and the time at which these results are produced.
- A soft real-time system is a system whose operation is degraded if results are not produced according to the specified timing requirements.
- A hard real-time system is a system whose operation is incorrect if results are not produced according to the timing specification.

Characteristics of embedded systems

- ✧ Embedded systems generally run continuously and do not terminate.
- ✧ Interactions with the system's environment are unpredictable.
- ✧ There may be physical limitations that affect the design of a system.
- ✧ Direct hardware interaction may be necessary.
- ✧ Issues of safety and reliability may dominate the system design.

State machine diagrams are help

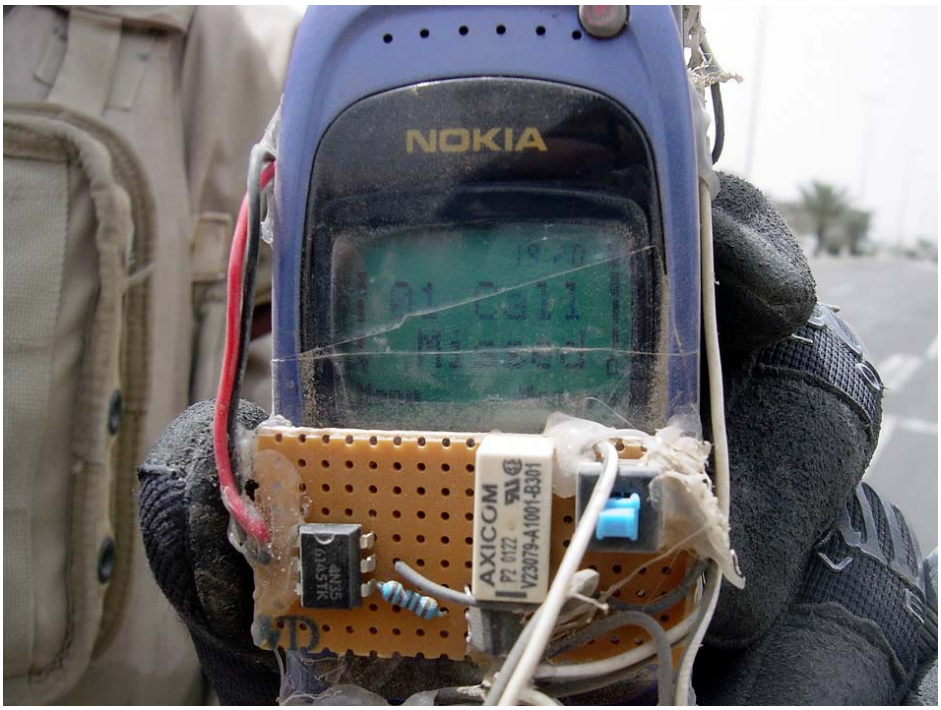
In embedded systems (and real-time systems, too) you should think about the order of different actions happening in the system.

State (transition) diagrams may help this a lot. There you draw possible state transitions. **Sometimes you have better make sure at code level, that unwished state transitions DO NOT happen.**

If your system is part of some larger system, remember that changes to the main system MAY affect the functionality (although it should not).

Even normal **updates sometimes break** something. E.g. mobile phones, copying machines, EXAM.

Is this an embedded system ?



Tampereen yliopisto
Tampere University

embedded software, e.g. cars

Traditionally, in old times, embedded systems were made hardware first (as it was slow development), and then software. A common misunderstanding was that ”hardware could be made almost working, as the functionality could be fixed easily with” software. Such happened e.g. at first F-18 fighter planes.

Nowadays you have better think, plan and design those in parallel.

For car software, earlier there were problems (not anymore ?) because (self-educated) ”car engineers” who had no ”sw dev training” started to write software.

For mobile phones, you may develop

- **native** application for a certain operating system, if you want quick response times and special features
- general **web** application, which runs on some browser (also at PC), if you want ”easy” development and wide use.

Tampereen yliopisto
Tampere University

Some cross-platform mobile dev tools

Xamarin is the preferred mobile app development tool for native applications. It reuses business logic layers and data access across platforms. It is widely used to build apps for iOS, Windows, and Android app development.

Appcelerator allows developers to create apps with fewer lines of code. This app development tool supports iOS, Android, Windows, and browser-based HTML5 applications.

PhoneGap is an Open Source free to use mobile app development framework. It falls into the category of cross-platform app development. It can be used for developing a single app which works on all mobile devices.

Ionic is HTML5 mobile app development framework. It is widely used for developing hybrid mobile apps. It is a useful tool to build mobile apps using web technologies like CSS, HTML5, and SASS.

Qt cross-platform SDK. It offers cost-effective design, development, and deployment. It allows developers to deliver the best user experience across all devices.

Alpha Anywhere is a rapid mobile app development and deployment tool. It is used for building cross platform web and mobile business apps.

[https://www.guru99.com/mobile-app-development-tools.html , 2019]

Summary

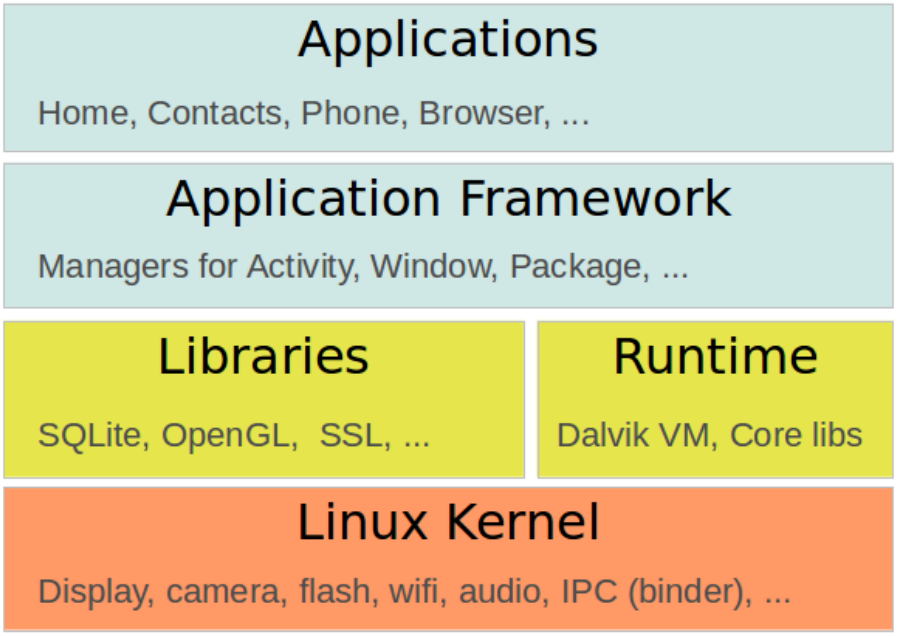
SIEMENS

The 7 Challenges of Embedded Software Development

1. **Consolidation**
   - shift from HW to SW
   - utilization of multi-/many-core systems
   - taking into account safety and real-time requirements
2. **Decentralization**
   - flexible deployment of functionality in distributed systems
3. **Heterogeneity**
   - heterogeneous multi-/many-core architectures
   - hardware accelerators
   - cloud computing
4. **Security**
   - data privacy
   - protection against manipulation
5. **Energy management**
   - power-efficient hard- and software
6. **Programming models**
   - Development efficiency and future-proofness
   - Portability, HW-independence
   - Scalability with processing power (more cores)
7. **Migration strategies**
   - utilize parallel hardware preserving existing code bases

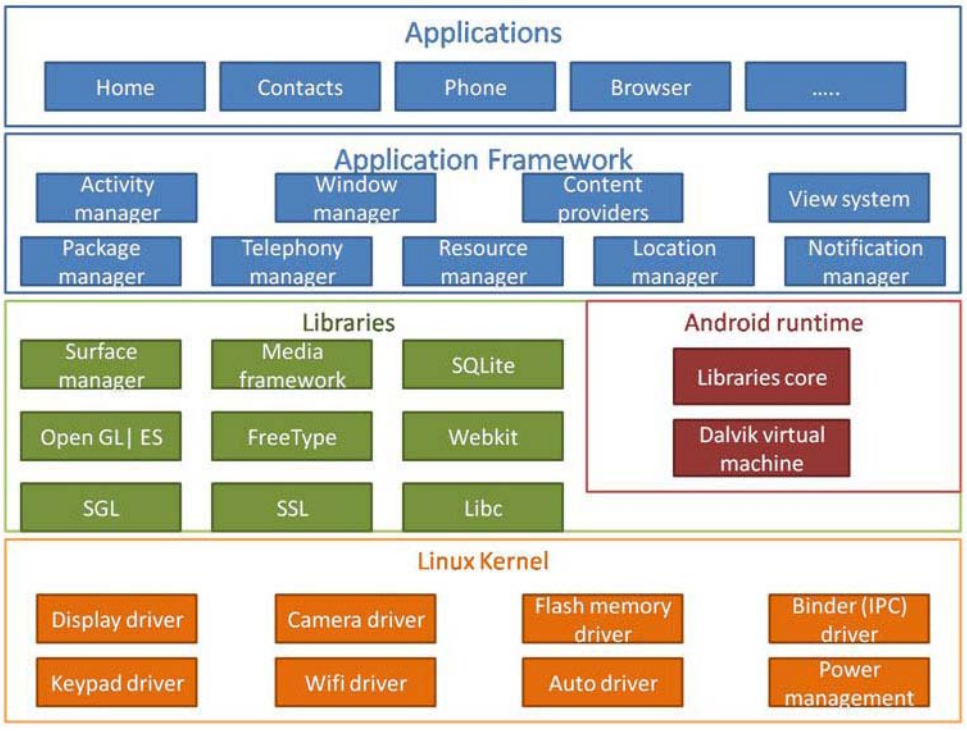
Seite 13 Januar 2012 Urs Gleim

Android software stack

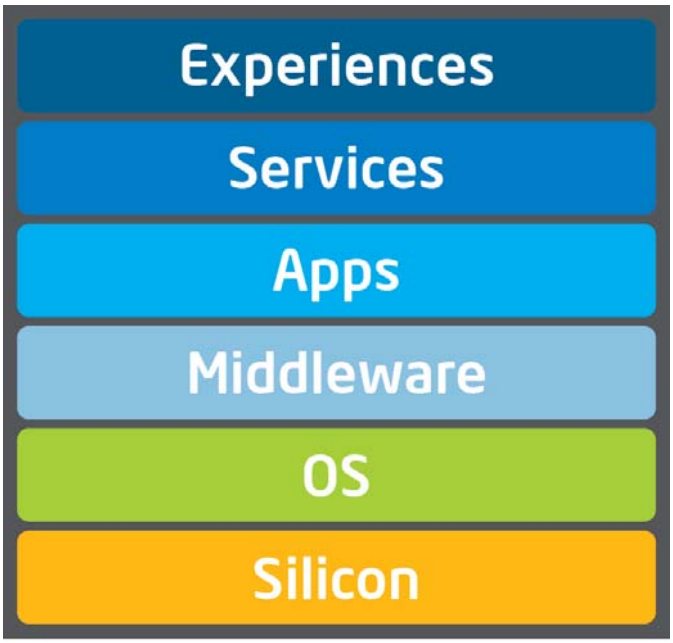


[https://dzone.com/articles/android-software-stack-and]

Android OS consists of different layers of software. Each layer groups several programs and each program has its own service to provide. Together, with other applications, these layers form the OS, middleware and applications.

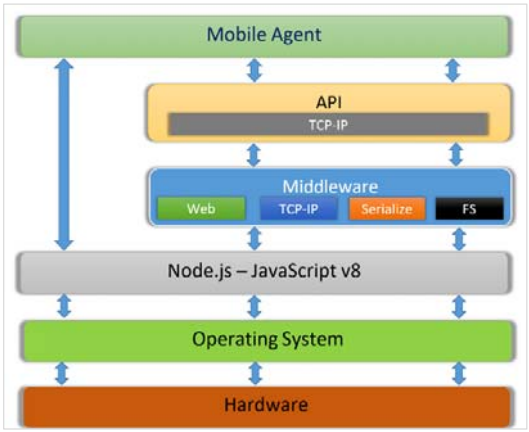


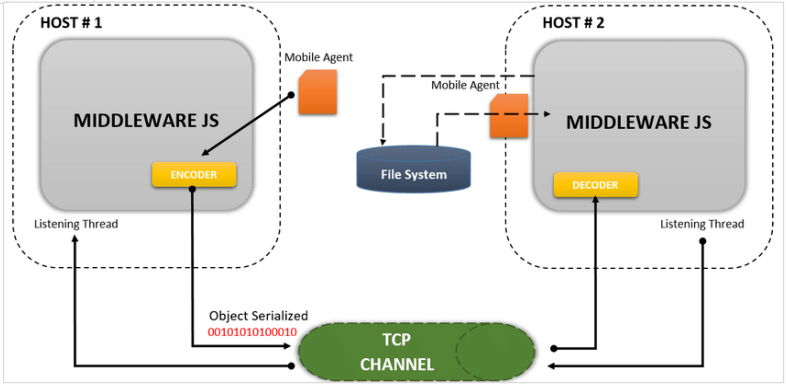
[http://www.learncertification.com/study-material/android-software-stack]



Adapted by Intel IT Center from Genevieve Bell's IDF 2013 Keynote Address

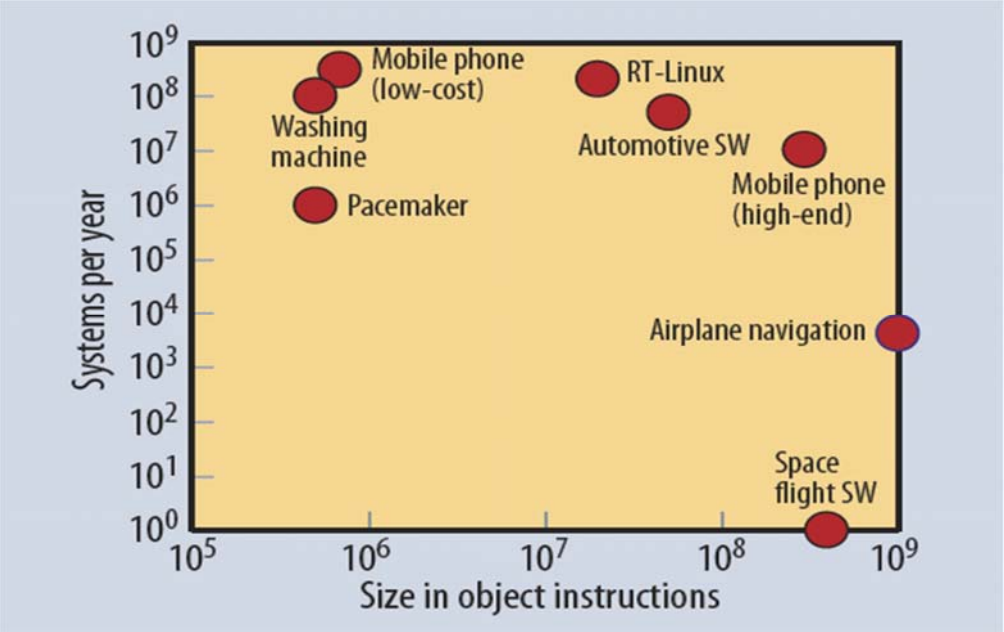
[https://itpeernetwork.intel.com/software-innovation-for-a-better-mobile-experience/#gs.h87q42]





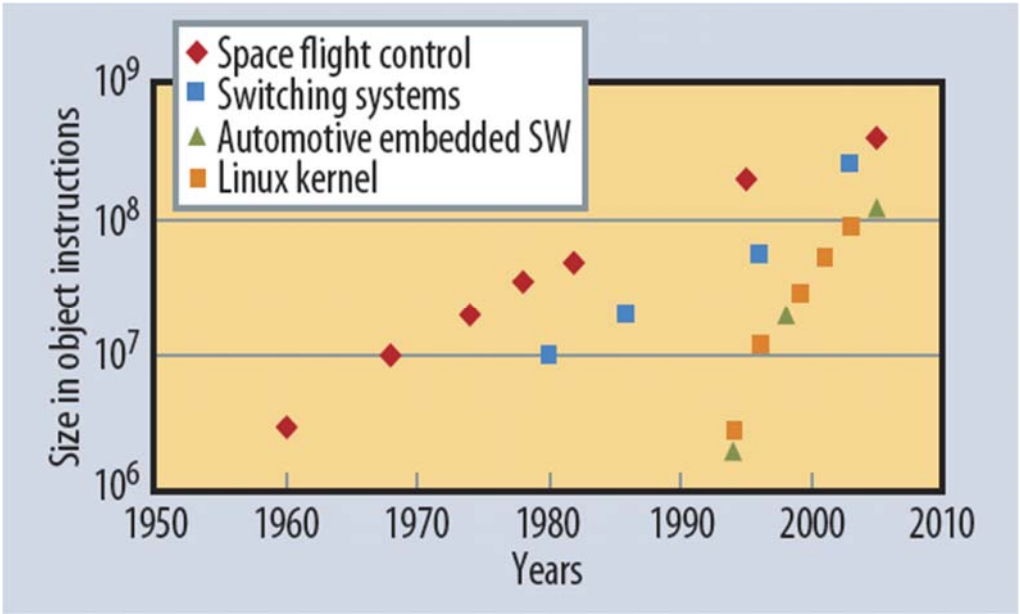
[Carlos Villafuerte: JavaScript Middleware for Mobile Agents Support on Desktop and Mobile Platforms, 2018]

Software size and complexity, increasing



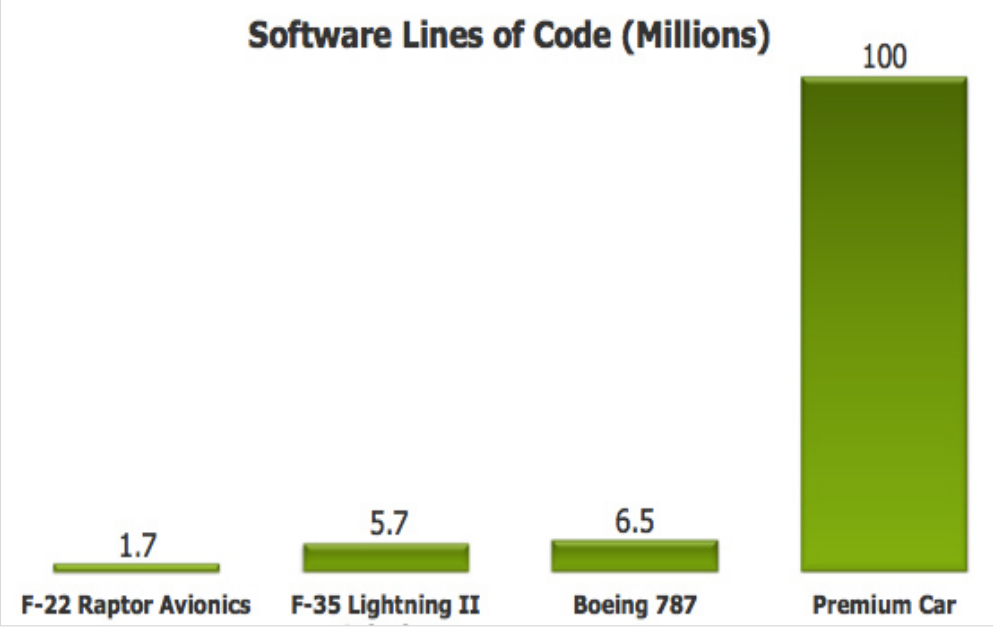
Embedded Software: Facts, Figures, and Future IEEE Computer, April 2009 (vol. 42 no. 4)

Software size, increasing all the time



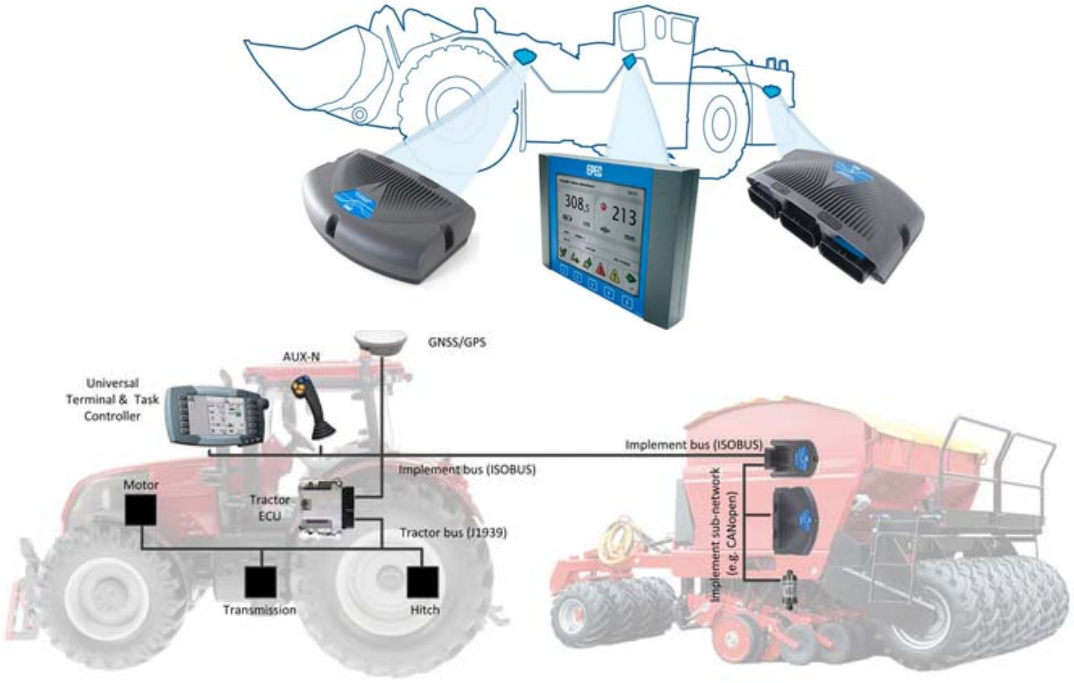
Embedded Software: Facts, Figures, and Future IEEE Computer, April 2009 (vol. 42 no. 4)

Some LOC counts

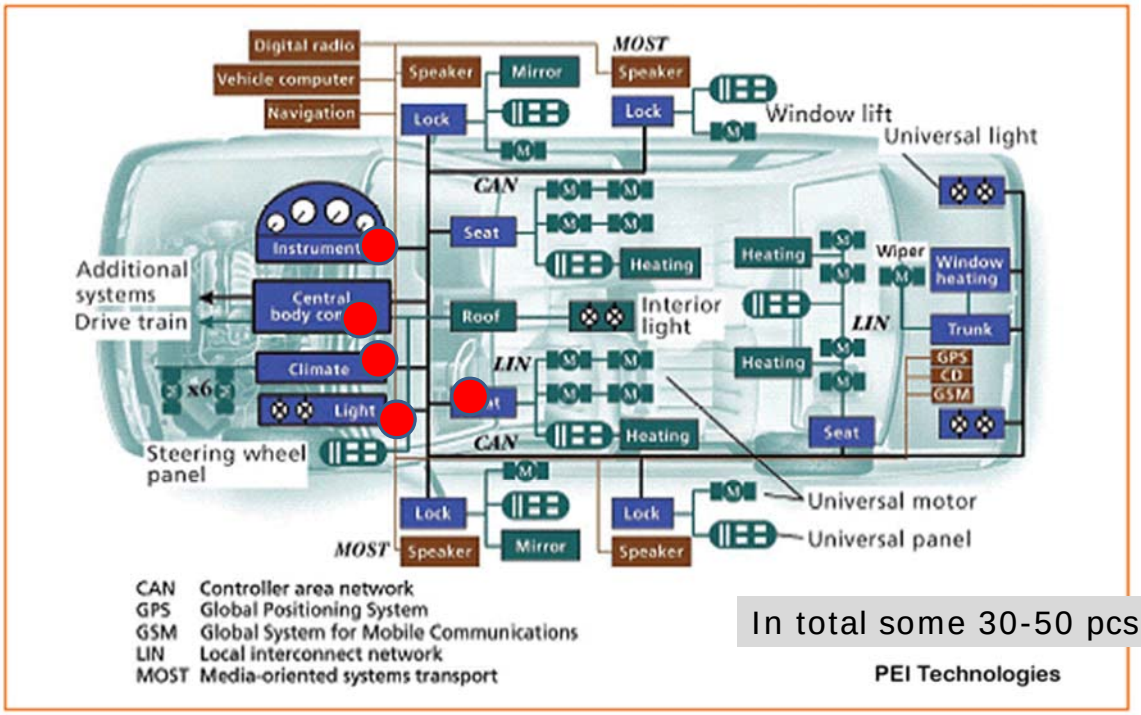


Modern car owner's pain…

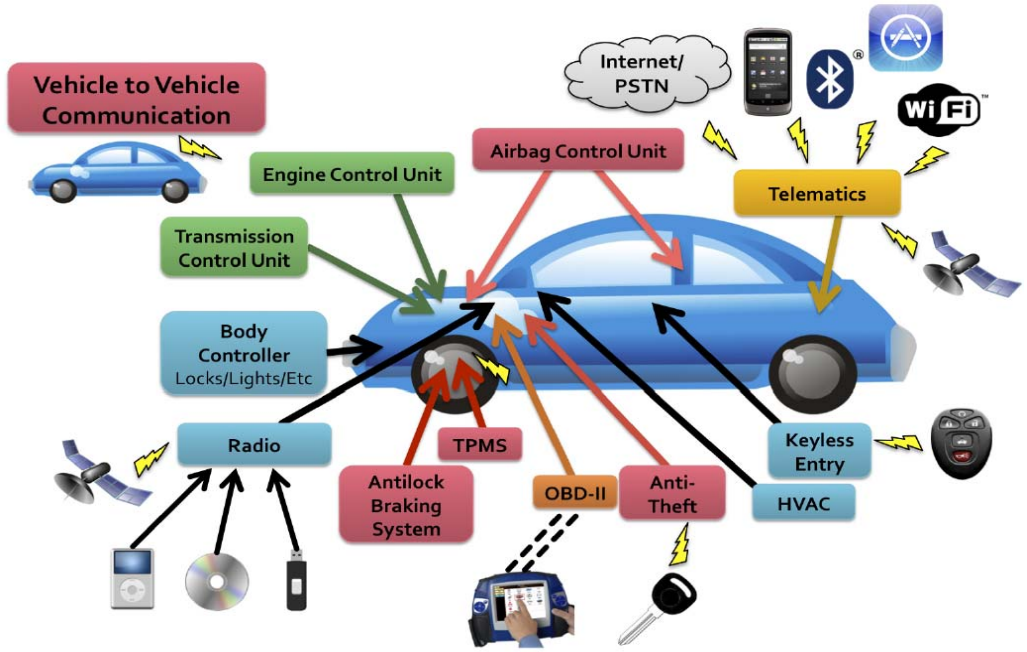
Architecture (ISOBUS)



http://www.aa1car.com/library/can_systems.htm



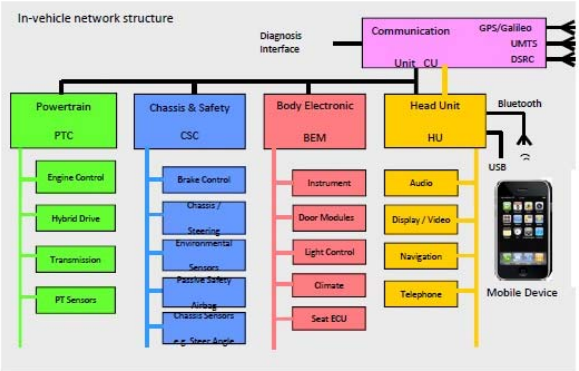
http://www.dailytech.com/Charlie+Miller+Releases+Open+Source+Car+Sabotage+Toolkit/article33308.htm



Security and safety risks in cars

Starting point – EVITA Use Cases 6

A suite of 18 potential use cases was defined, based on EASIS project network architecture



Scenario classes:

- car-car
- car-infrastructure
- mobile devices
- aftermarket
- maintenance

Assumed reference architecture

The Fully Networked Car
Geneva, 3-4 March 2010

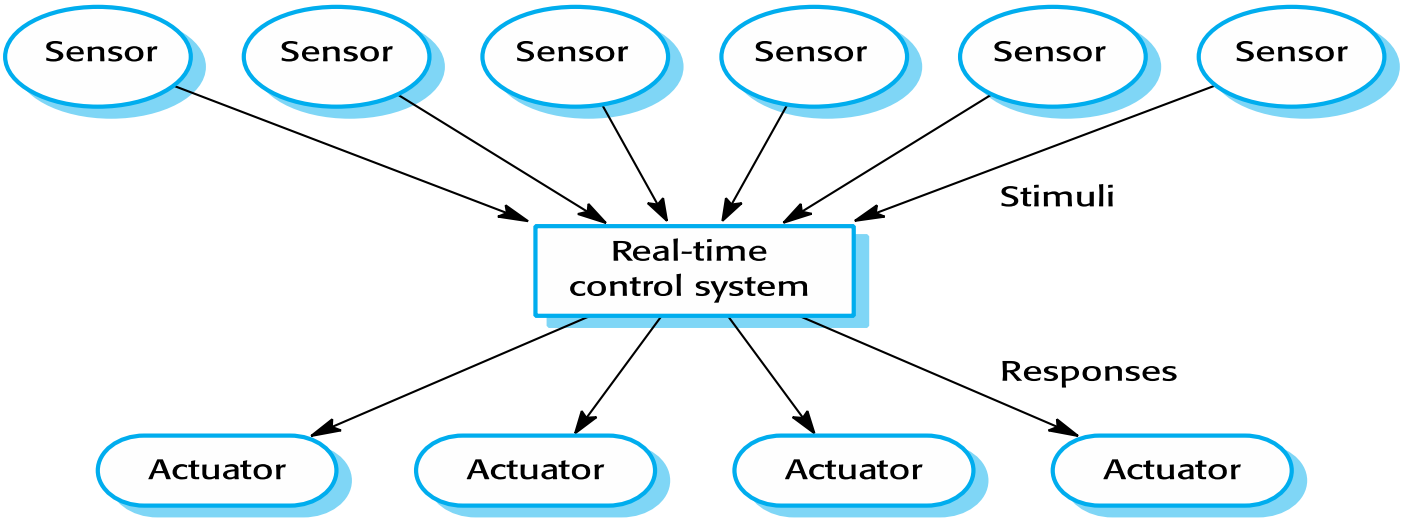
Embedded system design

- The design process for embedded systems is a systems engineering process that has to consider, in detail, the design and performance of the system hardware.
- Part of the design process may involve deciding which system capabilities are to be implemented in software and which in hardware.
- **Low-level decisions on hardware, support software and system timing must be considered early in the process.**
- These may mean that additional software functionality, such as battery and power management, has to be included in the system.

Reactive systems

- Real-time systems are often considered to be reactive systems. Given a stimulus, the system must produce a reaction or response within a specified time.
- Periodic stimuli. Stimuli which occur at predictable time intervals
  - For example, a temperature sensor may be polled 10 times per second.
- Aperiodic stimuli. Stimuli which occur at unpredictable times
  - For example, a system power failure may trigger an interrupt which must be processed by the system.

A general model of an embedded real-time system



Architectural considerations

- Because of the need to respond to timing demands made by different stimuli/responses, the system architecture must allow for fast switching between stimulus handlers.
- Timing demands of different stimuli are different so a simple sequential loop is not usually adequate.
- Real-time systems are therefore usually designed as cooperating processes with a real-time executive controlling these processes.

Design process activities

- ✧ Platform selection
- ✧ Stimuli/response identification
- ✧ Timing analysis
- ✧ Process design
- ✧ Algorithm design
- ✧ Data design
- ✧ Process scheduling.

F-22 Raptors' systems crash mid-flight over Pacific

Lockheed's shiny new F-22 Raptor stealth fighters may have owned a few war games, but **crossing the International Date Line** left them as helpless as a carrot in a rabbit trap, with **multiple system crashes** causing an emergency detour en route from Hawaii to Okinawa, Japan.

Communication, fuel subsystems, and navigation systems were rendered useless and repeated "reboots" were of no help. Luckily, the fleet had clear skies and refueling tankers to guide them back to Hawaii. If they had separated from the tankers, "they would have turned around and probably could have found the Hawaiian Islands. But if the weather had been bad on approach, there could have been real trouble," states Retired Air Force Major General Don Shepperd.

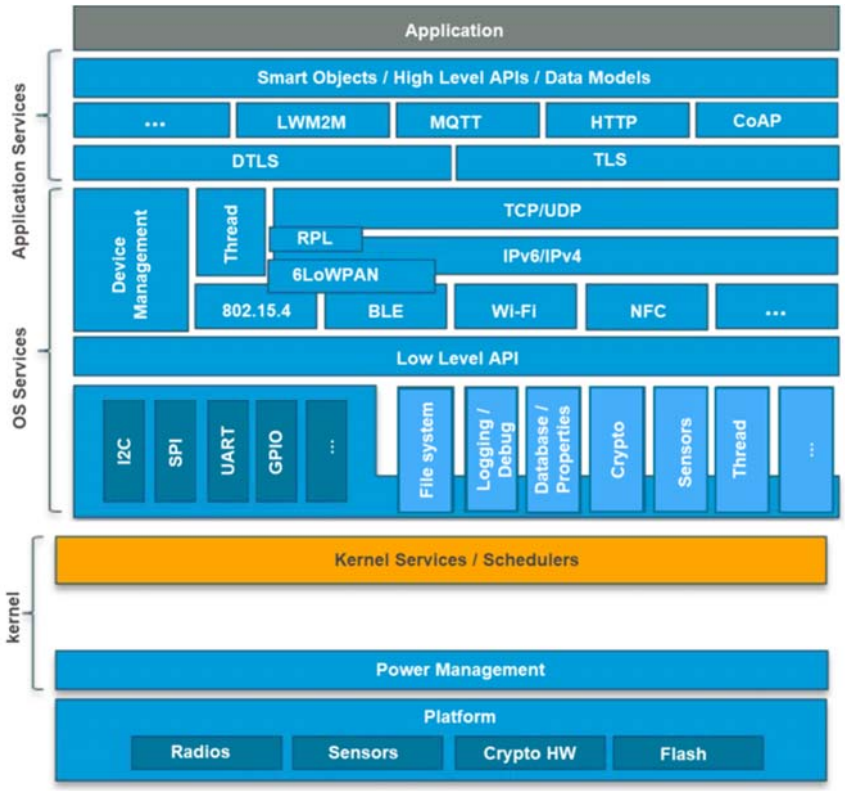
The voyage suffered a two-day delay on account of the system failures -- "a computer glitch in the millions of lines of code, **somebody made an error in a couple lines of the code and everything goes**." What should have been a showy parade of $125+ million super fighters quickly turned to disaster for Lockheed who would've had a lot of explaining to do, had this happened during combat.

"**suffered simultaneous total nav-console crashes as their longitude shifted from 180 degrees West to 180 East**"

[https://www.engadget.com/**2007**/02/27/f-22-raptors-systems-crash-mid-flight-over-pacific]

RT, Real-time systems

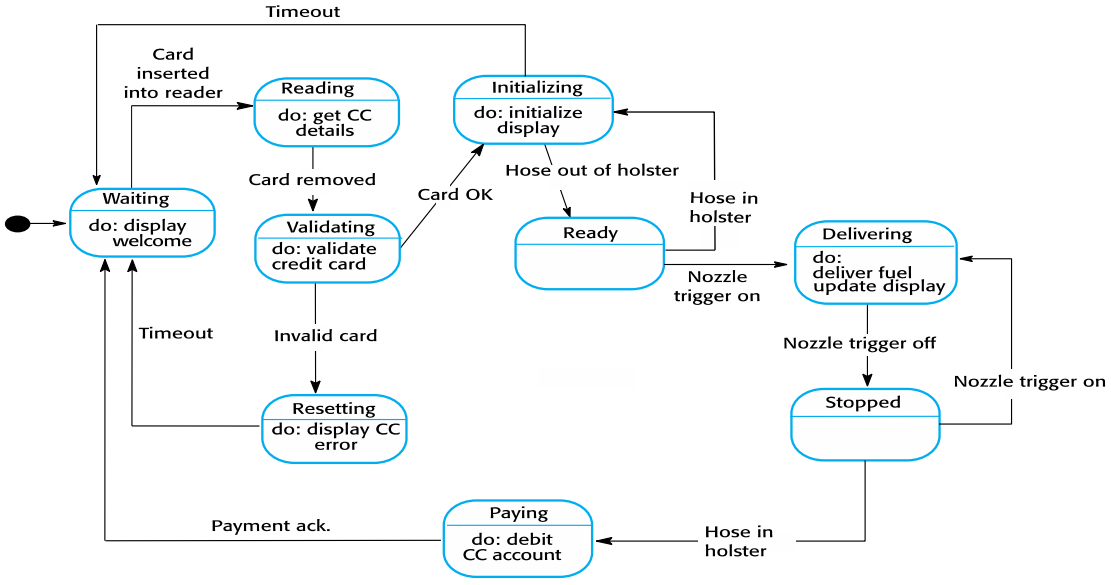
Zephyr, an Open Source RTOS for IoT



Real-time system modelling

- The effect of a stimulus in a real-time system may trigger a transition from one state to another.
- State models are therefore often used to describe embedded real-time systems.
- UML state diagrams may be used to show the states and state transitions in a real-time system.

State machine model of a petrol (gas) pump



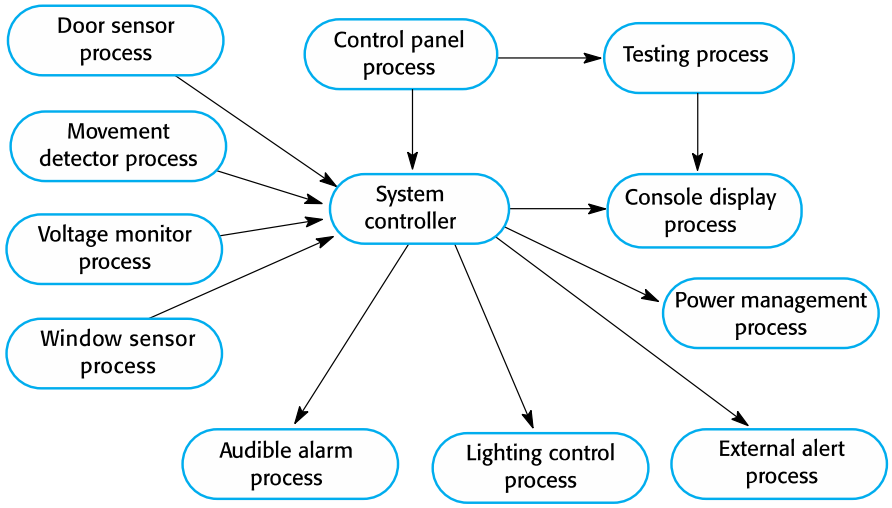
Sequence of actions in real-time pump control system

- The buyer inserts a credit card into a card reader built into the pump.
- Removal of the card triggers a transition to a Validating state where the card is validated.
- If the card is valid, the system initializes the pump and, when the fuel hose is removed from its holster, transitions to the Delivering state.
- After the fuel delivery is complete and the hose replaced in its holster, the system moves to a Paying state.
- After payment, the pump software returns to the Waiting state

Real-time programming

- Programming languages for real-time systems development have to include facilities to access system hardware, and it should be possible to predict the timing of particular operations in these languages.
- Systems-level languages, such as C, which allow efficient code to be generated are widely used in preference to languages such as Java.
- There is a performance overhead in object-oriented systems because extra code is required to mediate access to attributes and handle calls to operations. The loss of performance may make it impossible to meet real-time deadlines.

Process structure for a burglar alarm system



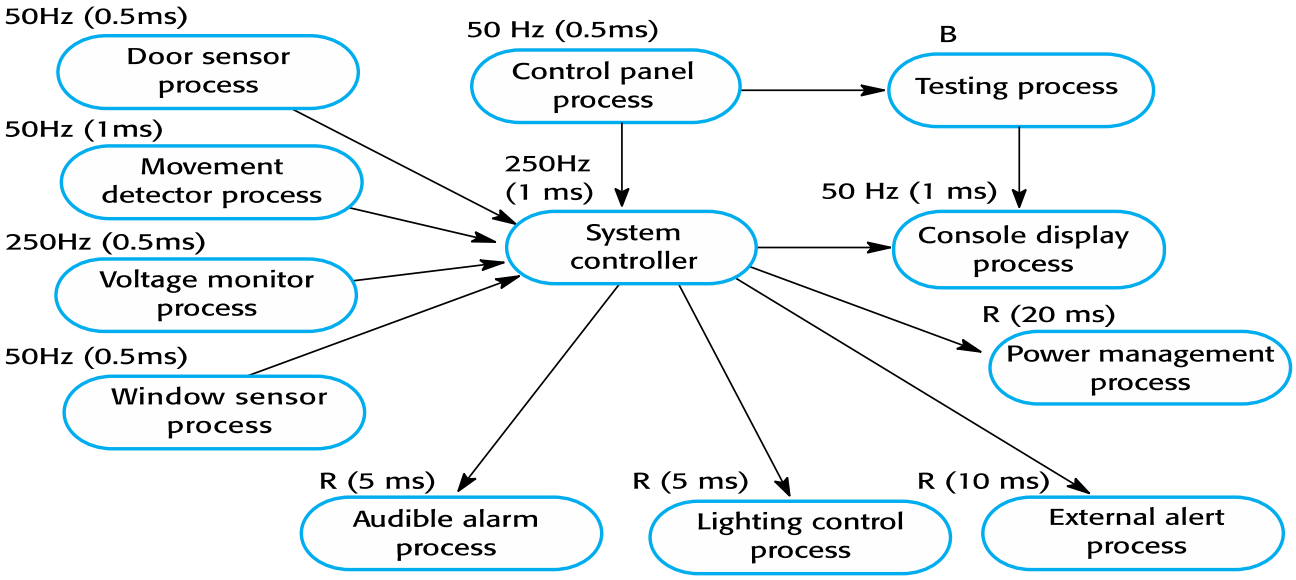
Timing analysis

- The correctness of a real-time system depends not just on the correctness of its outputs but also on the time at which these outputs were produced.
- In a timing analysis, you calculate how often each process in the system must be executed to ensure that all inputs are processed and all system responses produced in a timely way.
- The results of the timing analysis are used to decide how frequently each process should execute and how these processes should be scheduled by the real-time operating system.

Factors in timing analysis

- ***Deadlines***
  - The times by which stimuli must be processed and some response produced by the system.
- ***Frequency***
  - The number of times per second that a process must execute so that you are confident that it can always meet its deadlines.
- ***Execution time***
  - The time required to process a stimulus and produce a response.

Alarm process timing



Stimuli to be processed

- Power failure is detected by observing a voltage drop of more than 20%.
  - The required response is to switch the circuit to backup power by signalling an electronic power-switching device that switches the mains power to battery backup.
- Intruder alarm is a stimulus generated by one of the system sensors.
  - The response to this stimulus is to compute the room number of the active sensor, set up a call to the police, initiate the voice synthesizer to manage the call, and switch on the audible intruder alarm and building lights in the area.

Frequency and execution time

- The deadline for detecting a change of state is 0.25 seconds, which means that each sensor has to be checked 4 times per second. If you examine 1 sensor during each process execution, then if there are N sensors of a particular type, you must schedule the process 4N times per second to ensure that all sensors are checked within the deadline.
- If you examine 4 sensors, say, during each process execution, then the execution time is increased to about 4 ms, but you need only run the process N times/second to meet the timing requirement.

Key points

- **An embedded software system is part of a hardware/software system that reacts to events in its environment**. The software is 'embedded' in the hardware. Embedded systems are normally real-time systems.
- **A real-time system is a software system that must respond to events in real time**. System correctness does not just depend on the results it produces, but also on the time when these results are produced.
- Real-time systems are usually implemented as a set of communicating processes that react to stimuli to produce responses.
- **State models are an important design representation** for embedded real-time systems. They are used to show how the system reacts to its environment as events trigger changes of state in the system.

Key points

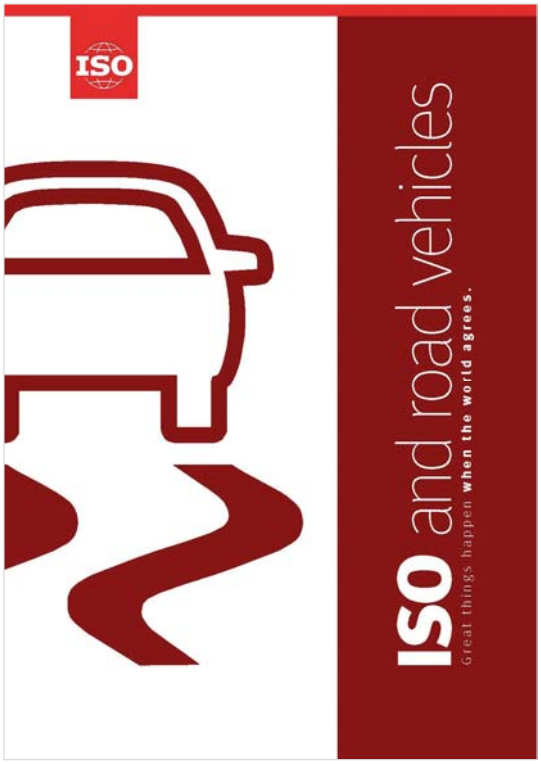
- There are several standard patterns that can be observed in different types of embedded system. These include a pattern for monitoring the system's environment for adverse events, a pattern for actuator control and a data-processing pattern.
- **Designers of real-time systems have to do a timing analysis**, which is driven by the deadlines for processing and responding to stimuli. They have to decide how often each process in the system should run and the expected and worst-case execution time for processes.
- **A real-time operating system is responsible for process and resource management**. It always includes a **scheduler**, which is the component responsible for deciding which process should be scheduled for execution.

safety

Software safety

- ”mission-critical” and ”safety-critical” software systems (space, aviation, marine, medical, military, law enforcement,…) have their own branch of safety standards

The **increasing complexity** of systems and the introduction and expansion of standards, codes and certification requirements has led to a cumulative demand for functional safety analyses in the high-tech automotive industry.



ISO 26262
Road vehicles — Functional safety

Is ISO 26262 just about software development?

ISO 26262 covers the entire product development lifecycle of electrical / electronic automotive products. The standard is composed of 10 parts, as shown below:

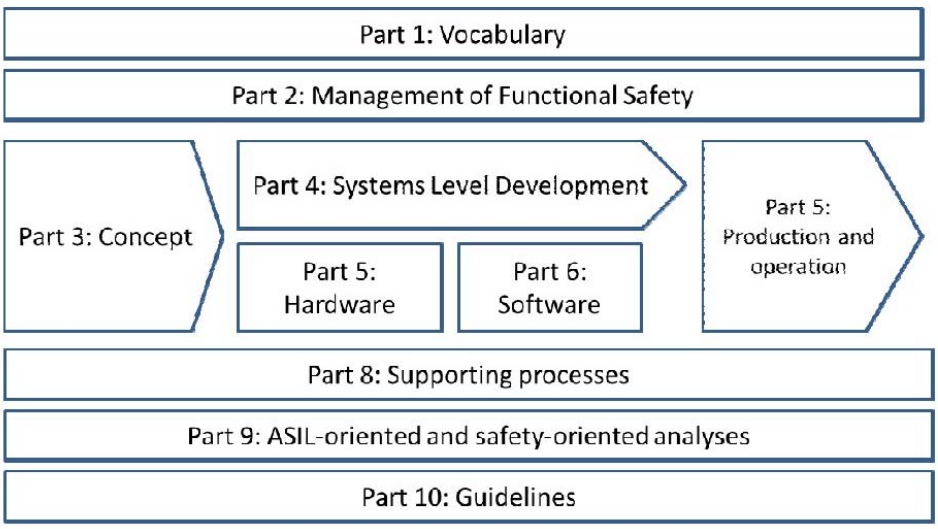
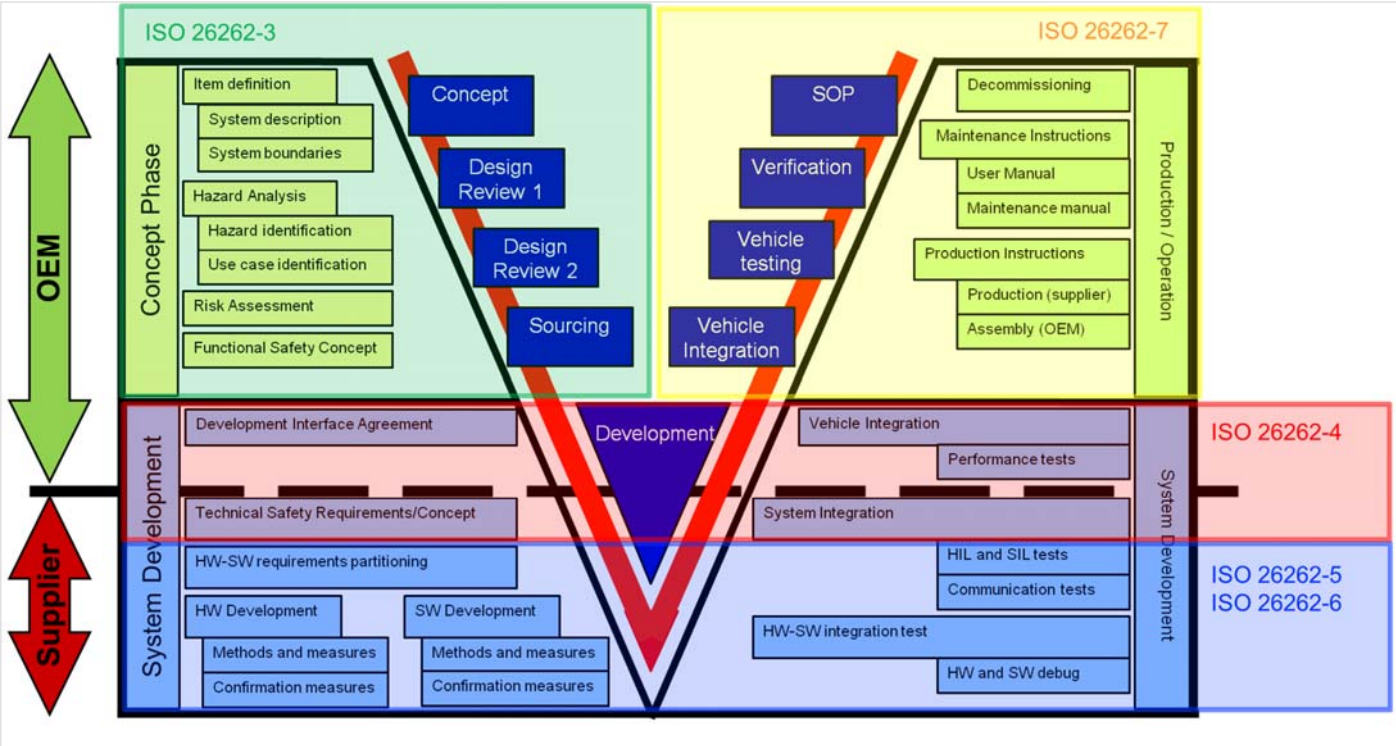


Figure 1 - The parts of ISO 26262

[www.feabhas.com]



[https://www.berns-engineers.com/be3/en/safety-engineering/functional-safety-automotive]

Four views on software systems safety

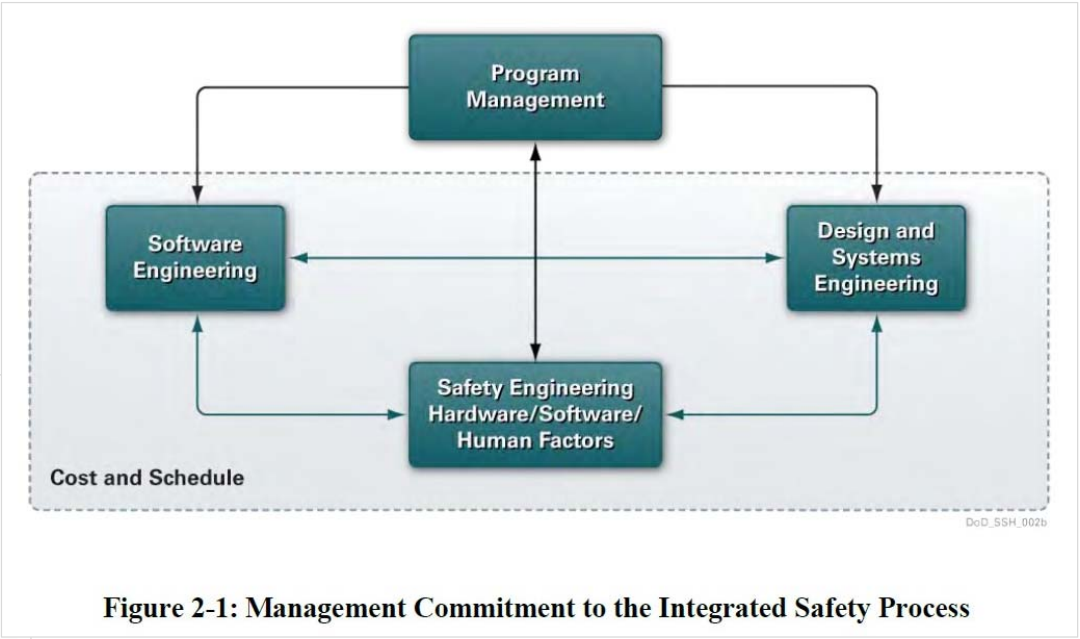
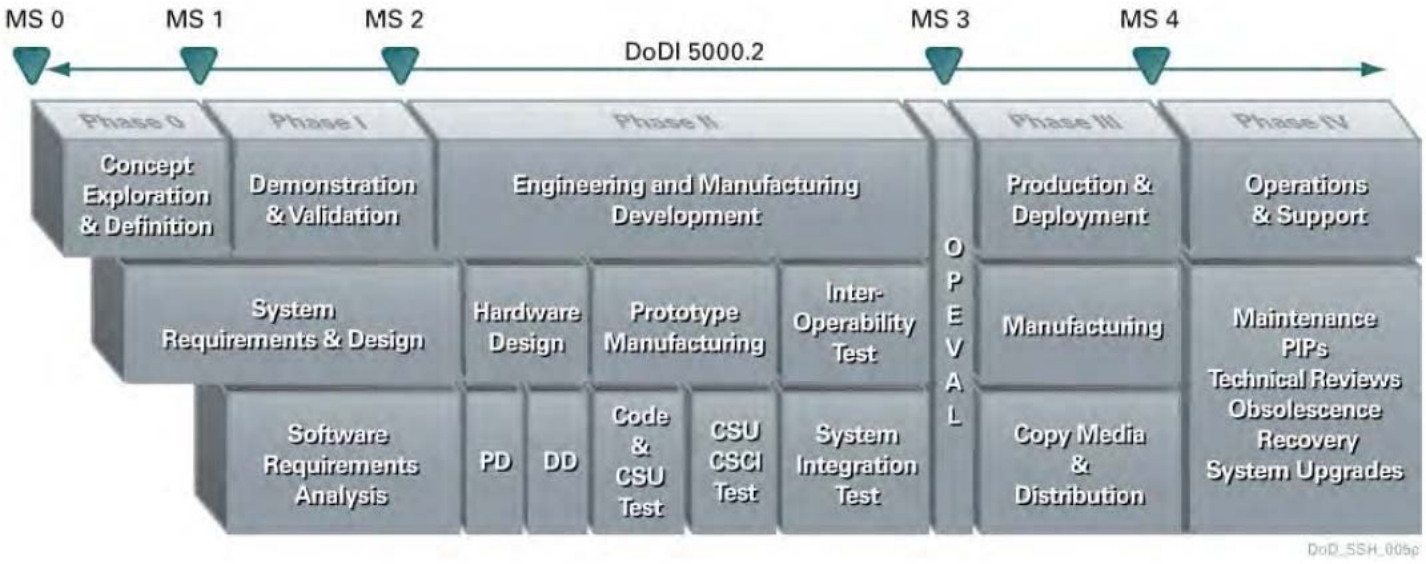


Figure 2-1: Management Commitment to the Integrated Safety Process

milestones



PD = program design, DD = detailed design, CSU = computer software unit, CSCI = computer software configuration item.

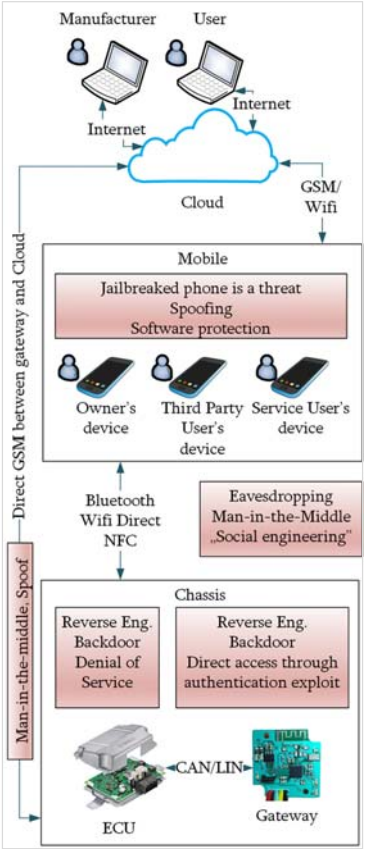
Figure 2-4: Relationship of Software to the Hardware Development Lifecycle

[Software System Safety Engineering Handbook, 2010]

[Barry Sheehan et al.: Connected and autonomous vehicles: A cyber-risk classificationframework, 2019]

Security issues

[https://www.researchgate.net/publication/281447339_Security_issues_and_vulnerabilities_in_connected_car_systems]



Driver Errors, Overreliance on Automation, Lack of Safeguards, Led to Fatal Tesla Crash

9/12/2017

WASHINGTON (Sept. 12, 2017) — The National Transportation Safety Board determined Tuesday that a truck driver's failure to yield the right of way and a **car driver's inattention** due to **overreliance on vehicle automation** are the probable cause of the fatal May 7, 2016, crash near Williston, Florida.

The NTSB also determined the operational design of the Tesla's vehicle automation permitted the car driver's overreliance on the automation, noting its design allowed prolonged disengagement from the driving task and enabled the driver to use it in ways inconsistent with manufacturer guidance and warnings.

As a result of its investigation the NTSB issued seven new safety recommendations and reiterated two previously issued safety recommendations.

"Neither Autopilot nor the driver noticed the white side of the tractor trailer against a brightly lit sky, so the brake was not applied," Tesla said.

Business Insider, Cadie Thompson
Jun 20, 2017, 6:06 PM

Somebody has set up a web page
https://www.tesladeaths.com/

Los Angeles Times

Tesla car was on Autopilot when it hit a Culver City firetruck, NTSB finds

A Tesla car on Autopilot rear-ended a Culver City firetruck on the 405 Freeway on Jan. 22, 2018. The firetruck was stopped with its lights flashing as firefighters attended to a previous crash there. (Culver City Firefighters Local 1927)

By ASSOCIATED PRESS SEP. 3, 2019 | 4:20 PM

A government report says the driver of a Tesla sedan that slammed into a Culver City firetruck on the 405 Freeway last year was using the car's Autopilot system when a

NATIONAL TRANSPORTATION SAFETY BOARD

Search this site... Search Site
Advanced Search

HOME NEWS & EVENTS SAFETY ADVOCACY INVESTIGATIONS DISASTER ASSISTANCE LEGAL ABOUT SEARCH NEW
PUBLICATIONS

Home > NEWS & EVENTS > News Releases > Driver Errors, Advanced Driver Assistance System Design, Led to Highway Crash

SHARE

NTSB News Release

National Transportation Safety Board Office of Public Affairs

Driver Errors, Advanced Driver Assistance System Design, Led to Highway Crash

9/4/2019

WASHINGTON (Sept. 4, 2019) — A driver's inattention, overreliance on his car's advanced driver assistance system, and use of the system inconsistent with manufacturer guidance, coupled with the system permitting driver disengagement from the driving task, led to the Jan. 22, 2018, crash in Culver City, California, according to a National Transportation Safety Board brief issued Wednesday.

Highway Accident Brief 19/07 details the results of the NTSB's investigation of the crash involving a Tesla Model S P85 and a Culver City Fire Department 2006 Seagrave Fire Truck in the high-occupancy vehicle lane of southbound Interstate 405. No one was injured as a result of the crash.

The response to a collision in the northbound freeway lanes 25 minutes earlier left a California Highway Patrol vehicle parked on the left shoulder of southbound I-405 and the Culver City Fire Department truck parked diagonally across the southbound HOV lane. Emergency lights were active on both vehicles. The Tesla, which had its "Autopilot" system engaged, was traveling in the HOV lane, behind another vehicle. After the lead vehicle changed lanes to the right, the Tesla remained in the HOV lane, accelerated and struck the rear of the fire truck at a recorded speed of about 31 mph. A forward collision warning alert occurred 0.49 seconds prior to impact but the automatic emergency braking system did not engage. The driver's hands were not detected on the steering wheel during

Related News Releases
- September 04, 2019
  Driver Errors, Advanced Driver Assistance System Design, Led to Highway Crash

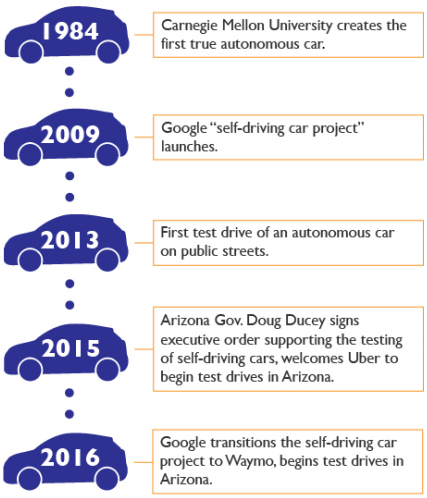
Related Reports
- Highway Accident Brief: Rear-End Collision Between a Car Operating with Advanced Driver Assistance Systems and a Stationary Fire Truck

Related Events

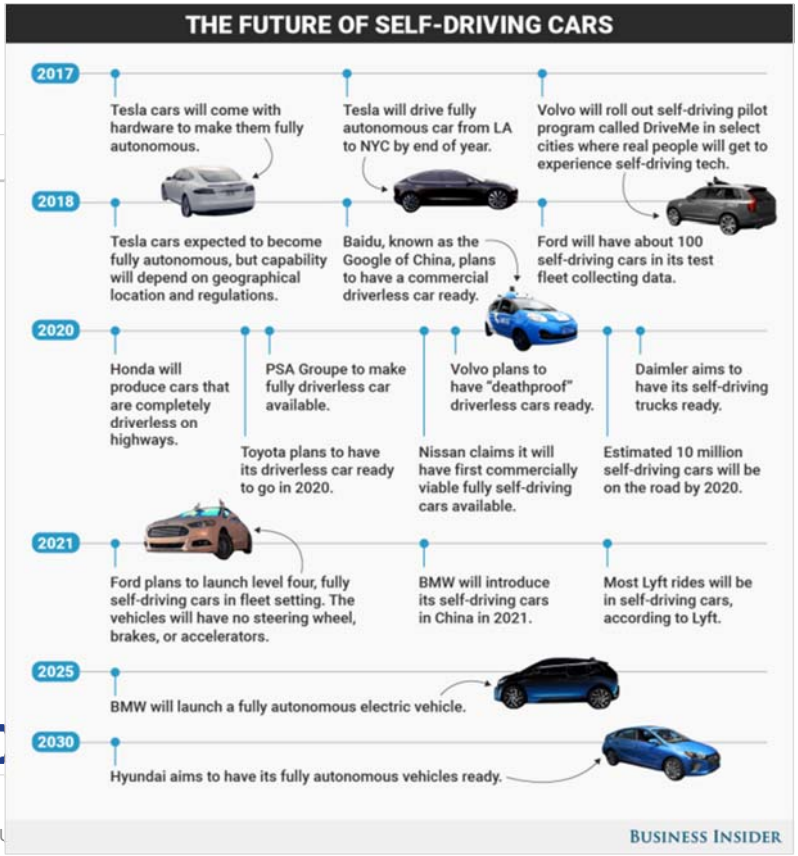
Related Investigations
- Car with automated vehicle controls crashes into fire truck

More NTSB Links
- Investigation Process
- Data & Stats
- Accident Reports
- Most Wanted List

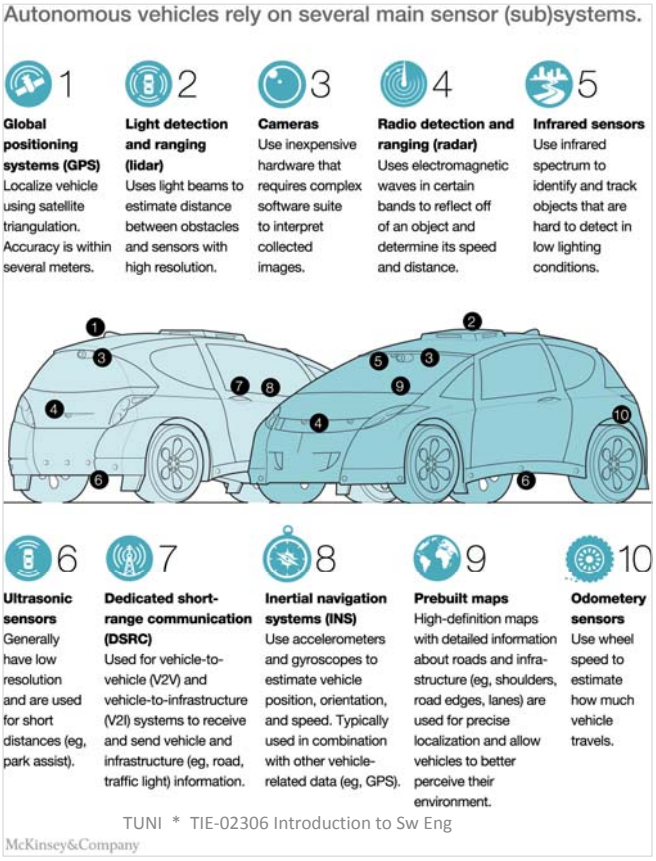
Timeline of self-driving cars

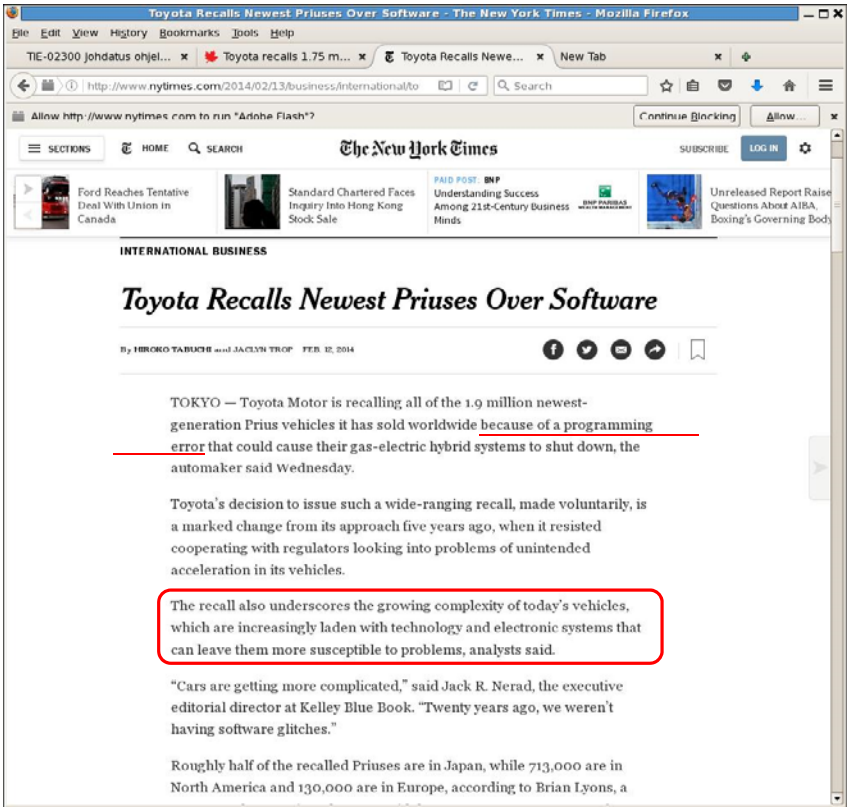


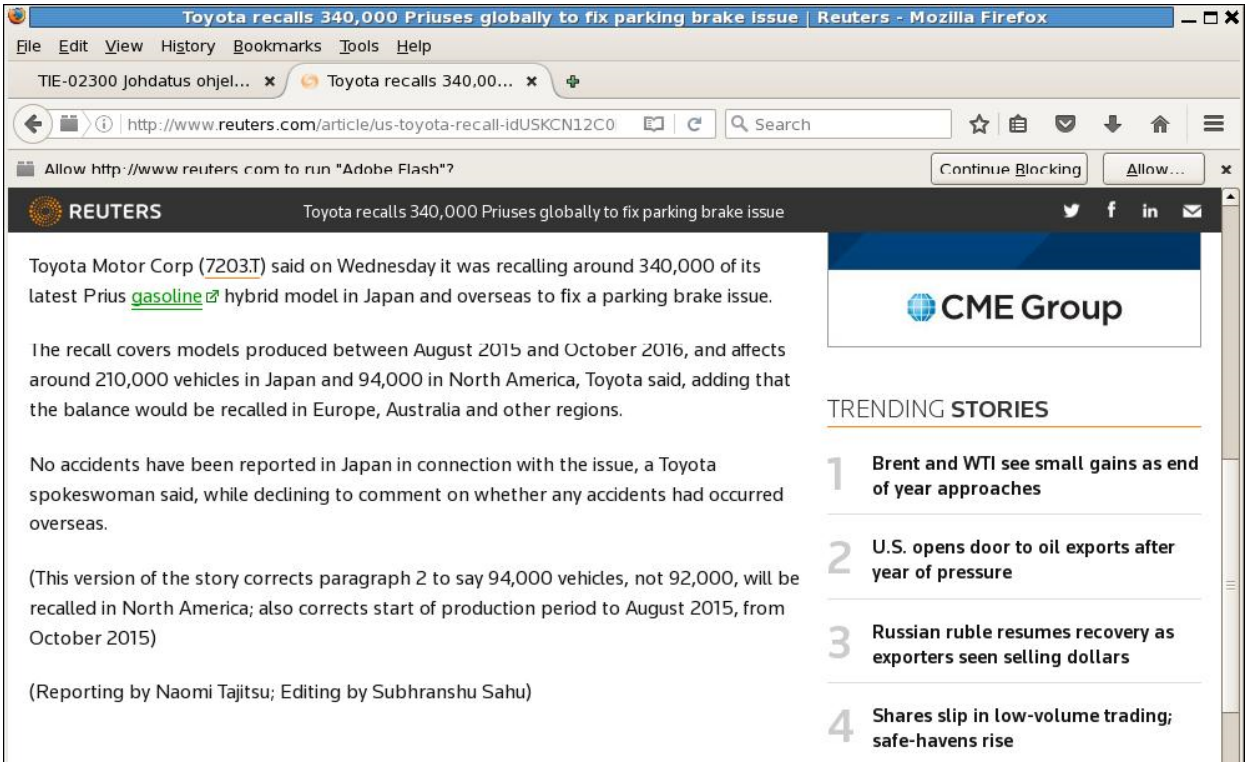
Graphic by Kayla Becton/Cronkite News

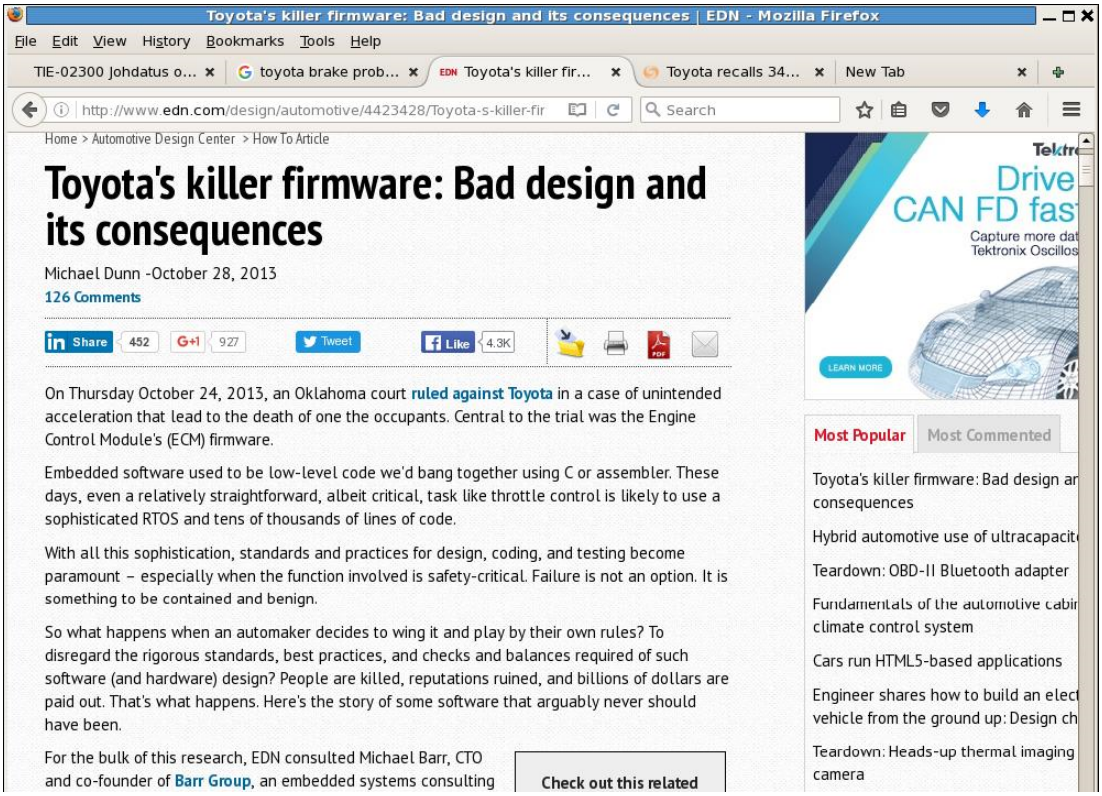


Example vehicle sensor systems

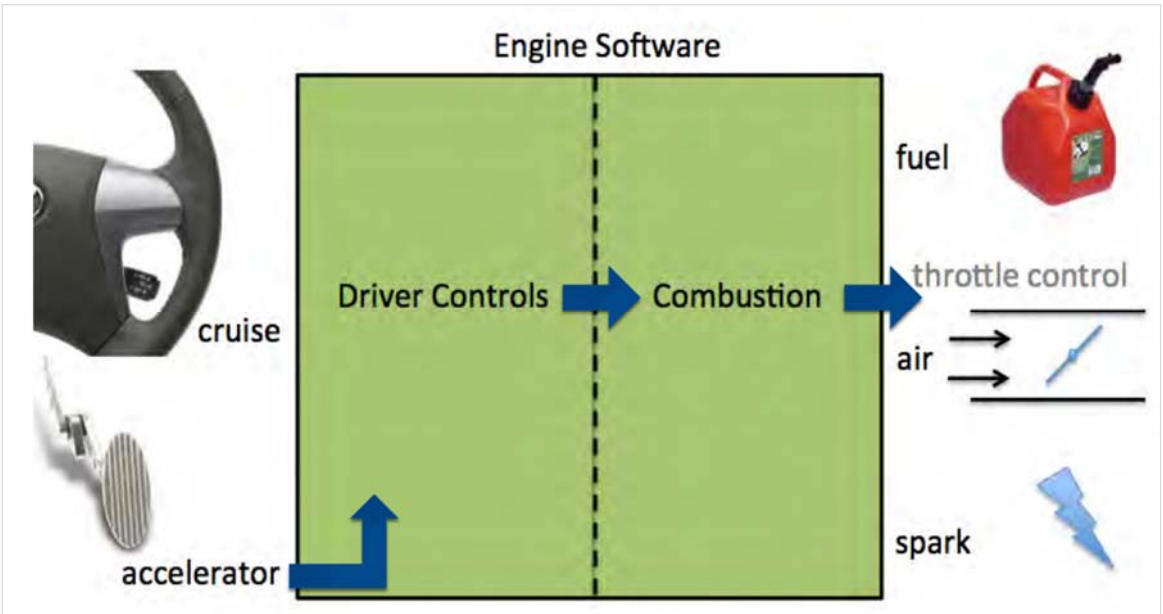



Toyota Recalls Newest Priuses Over Software - The New York Times - Mozilla Firefox
File Edit View History Bookmarks Tools Help
TIE-02300 Johdatus ohjel...
Toyota recalls 1.75 m...
Toyota Recalls Newe...
New Tab
http://www.nytimes.com/2014/02/13/business/international/to
Search
Allow http://www.nytimes.com to run "Adobe Flash"?
Continue Blocking
Allow...
SECTIONS
HOME
SEARCH
The New York Times
SUBSCRIBE
LOG IN
Ford Reaches Tentative Deal With Union in Canada
Standard Chartered Faces Inquiry Into Hong Kong Stock Sale
PAID POST: BNP
Understanding Success Among 21st-Century Business Minds
Unreleased Report Raise Questions About AIBA, Boxing's Governing Body
INTERNATIONAL BUSINESS
Toyota Recalls Newest Priuses Over Software
By HIROKO TABUCHI and JACLYN TROP FEB. 12, 2014
TOKYO — Toyota Motor is recalling all of the 1.9 million newest-generation Prius vehicles it has sold worldwide because of a programming error that could cause their gas-electric hybrid systems to shut down, the automaker said Wednesday.
Toyota's decision to issue such a wide-ranging recall, made voluntarily, is a marked change from its approach five years ago, when it resisted cooperating with regulators looking into problems of unintended acceleration in its vehicles.
The recall also underscores the growing complexity of today's vehicles, which are increasingly laden with technology and electronic systems that can leave them more susceptible to problems, analysts said.
"Cars are getting more complicated," said Jack R. Nerad, the executive editorial director at Kelley Blue Book. "Twenty years ago, we weren't having software glitches."
Roughly half of the recalled Priuses are in Japan, while 713,000 are in North America and 130,000 are in Europe, according to Brian Lyons, a

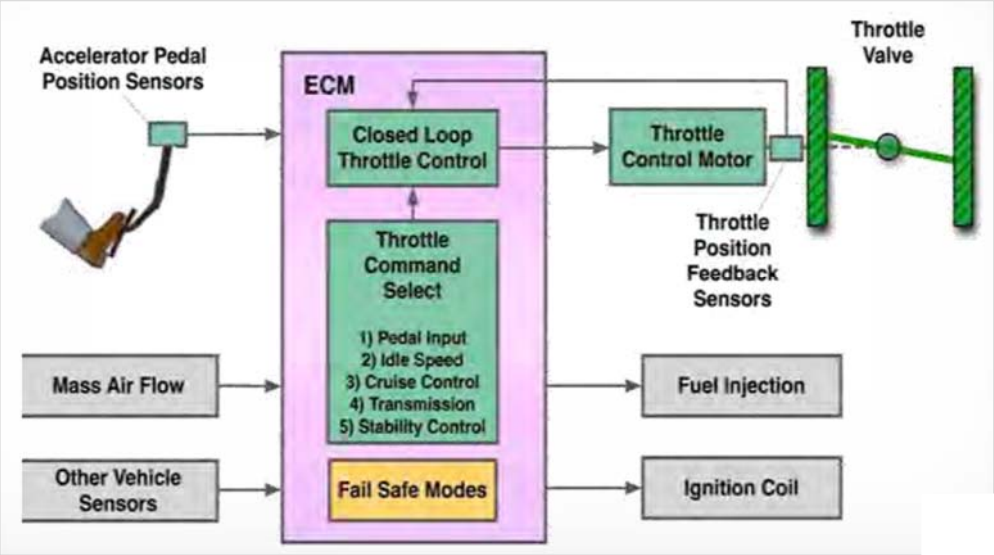

Toyota recalls 340,000 Priuses globally to fix parking brake issue | Reuters - Mozilla Firefox
File Edit View History Bookmarks Tools Help
TIE-02300 Johdatus ohjel...
Toyota recalls 340,00...
http://www.reuters.com/article/us-toyota-recall-idUSKCN12C0
Search
Allow http://www.reuters.com to run "Adobe Flash"?
Continue Blocking
Allow...
REUTERS
Toyota recalls 340,000 Priuses globally to fix parking brake issue
Toyota Motor Corp (7203.T) said on Wednesday it was recalling around 340,000 of its latest Prius gasoline hybrid model in Japan and overseas to fix a parking brake issue.
The recall covers models produced between August 2015 and October 2016, and affects around 210,000 vehicles in Japan and 94,000 in North America, Toyota said, adding that the balance would be recalled in Europe, Australia and other regions.
No accidents have been reported in Japan in connection with the issue, a Toyota spokeswoman said, while declining to comment on whether any accidents had occurred overseas.
(This version of the story corrects paragraph 2 to say 94,000 vehicles, not 92,000, will be recalled in North America; also corrects start of production period to August 2015, from October 2015)
(Reporting by Naomi Tajitsu; Editing by Subhranshu Sahu)
CME Group
TRENDING STORIES
1 Brent and WTI see small gains as end of year approaches
2 U.S. opens door to oil exports after year of pressure
3 Russian ruble resumes recovery as exporters seen selling dollars
4 Shares slip in low-volume trading; safe-havens rise



(http://www.safetyresearch.net/Library/BarrSlides_FINAL_SCRUBBED.pdf)



Same source (Toyota break case)



ECM = Engine Control Module

Hackers have self-driving cars in their headlights

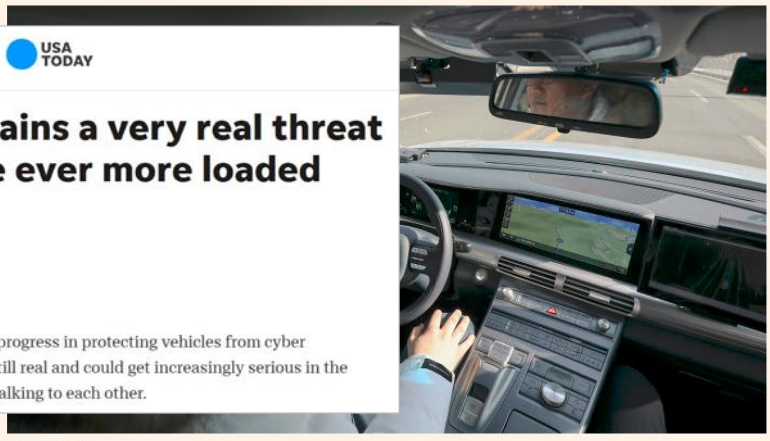
Greater connectivity gives criminals more access

USA TODAY

Car hacking remains a very real threat as autos become ever more loaded with tech

JC Reindl | Detroit Free Press
Published 1:56 PM EST Jan 15, 2018

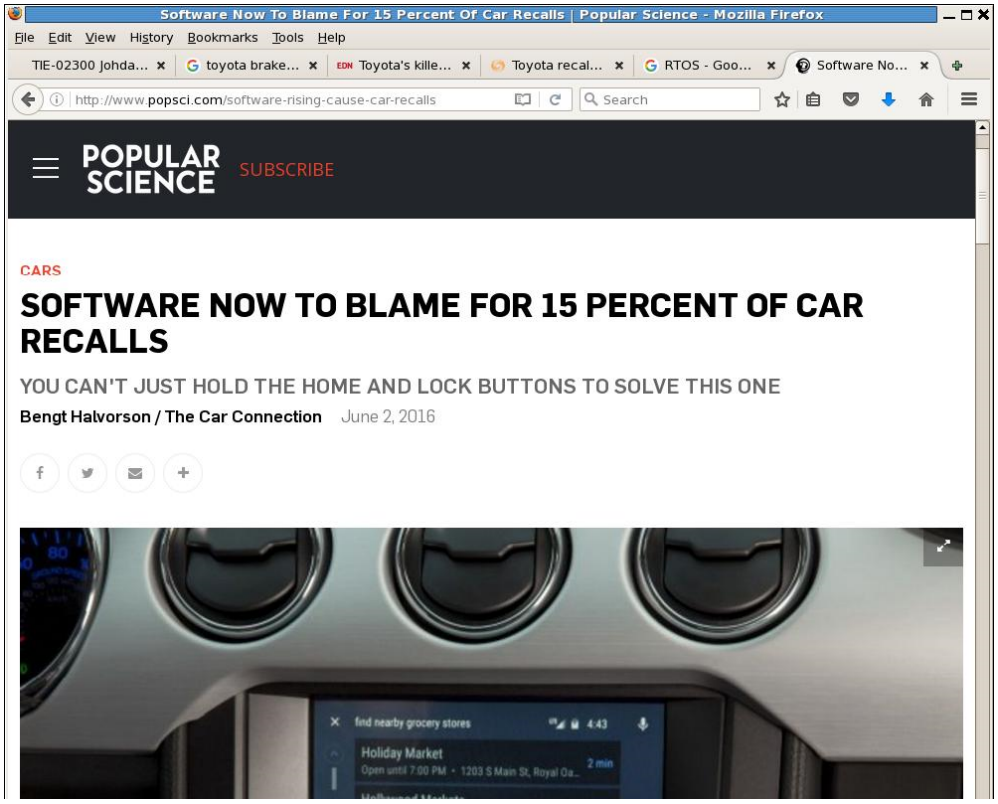
Automakers and suppliers are making progress in protecting vehicles from cyber attacks, but the car-hacking threat is still real and could get increasingly serious in the future when driverless vehicles begin talking to each other.

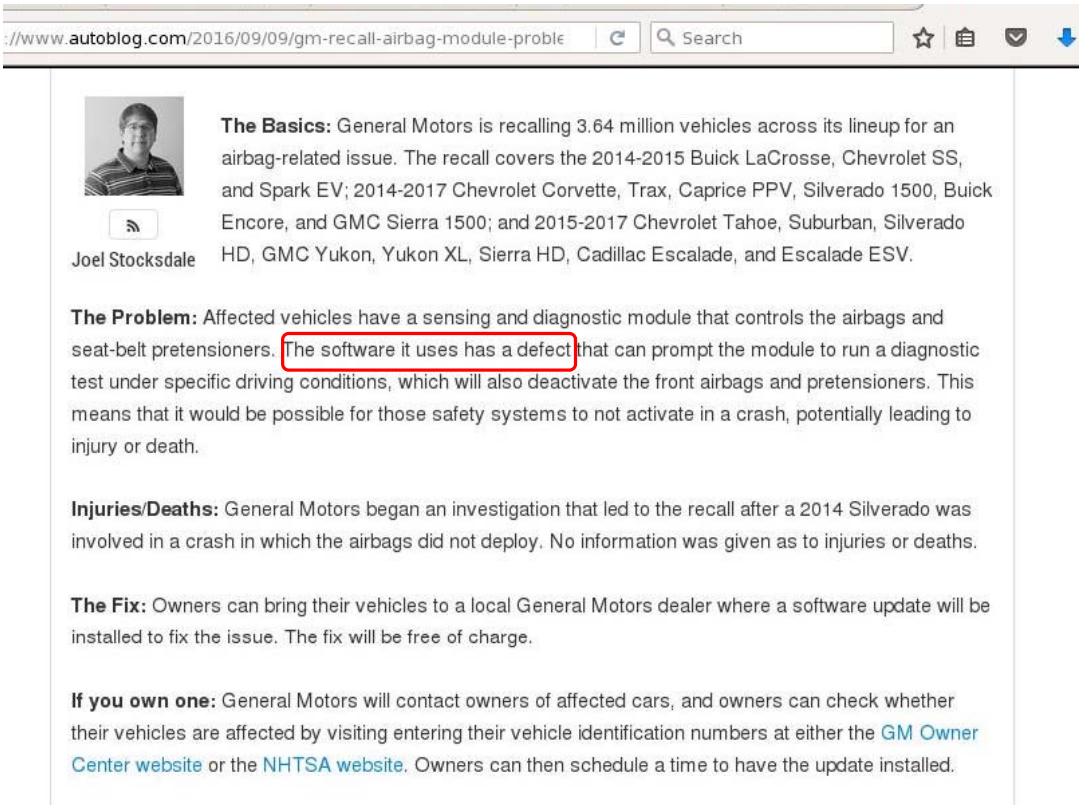


© Bloomberg

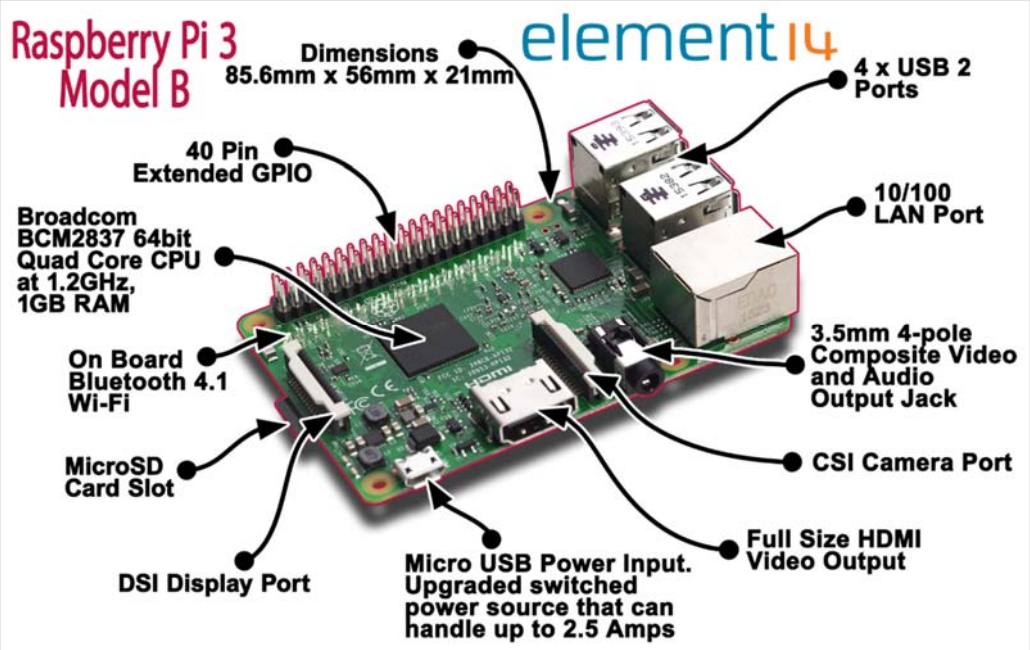
Peter Campbell, Motor Industry correspondent MARCH 15, 2018 11

With a crunch, the Jeep Cherokee rolled out of the car park and into the grassy ditch. But the terrified person in the driver's seat, a journalist from Wired, was

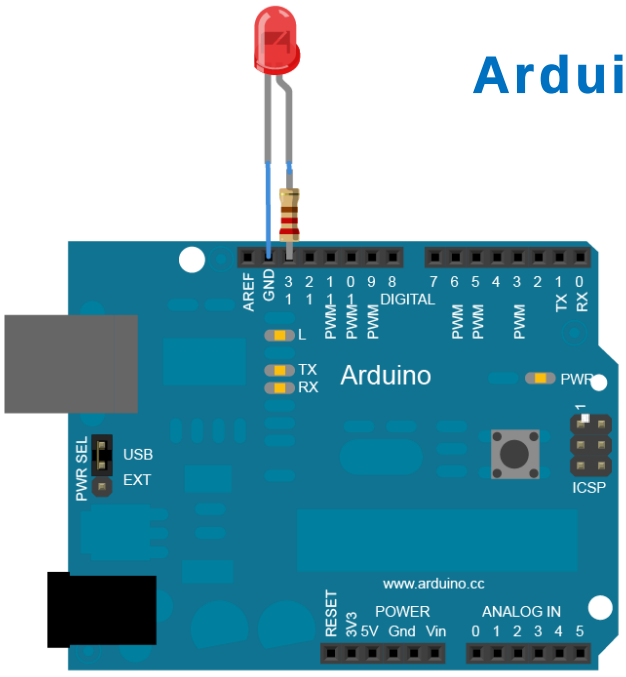

Software Now To Blame For 15 Percent Of Car Recalls | Popular Science - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.popsci.com/software-rising-cause-car-recalls
POPULAR SCIENCE
SUBSCRIBE
CARS
SOFTWARE NOW TO BLAME FOR 15 PERCENT OF CAR RECALLS
YOU CAN'T JUST HOLD THE HOME AND LOCK BUTTONS TO SOLVE THIS ONE
Bengt Halvorson / The Car Connection June 2, 2016
find nearby grocery stores
Holiday Market

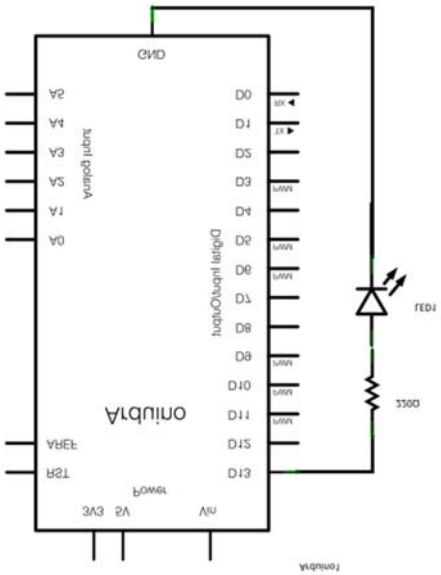

://www.autoblog.com/2016/09/09/gm-recall-airbag-module-proble
Search
Joel Stocksdale
The Basics: General Motors is recalling 3.64 million vehicles across its lineup for an airbag-related issue. The recall covers the 2014-2015 Buick LaCrosse, Chevrolet SS, and Spark EV; 2014-2017 Chevrolet Corvette, Trax, Caprice PPV, Silverado 1500, Buick Encore, and GMC Sierra 1500; and 2015-2017 Chevrolet Tahoe, Suburban, Silverado HD, GMC Yukon, Yukon XL, Sierra HD, Cadillac Escalade, and Escalade ESV.
The Problem: Affected vehicles have a sensing and diagnostic module that controls the airbags and seat-belt pretensioners. The software it uses has a defect that can prompt the module to run a diagnostic test under specific driving conditions, which will also deactivate the front airbags and pretensioners. This means that it would be possible for those safety systems to not activate in a crash, potentially leading to injury or death.
Injuries/Deaths: General Motors began an investigation that led to the recall after a 2014 Silverado was involved in a crash in which the airbags did not deploy. No information was given as to injuries or deaths.
The Fix: Owners can bring their vehicles to a local General Motors dealer where a software update will be installed to fix the issue. The fix will be free of charge.
If you own one: General Motors will contact owners of affected cars, and owners can check whether their vehicles are affected by visiting entering their vehicle identification numbers at either the GM Owner Center website or the NHTSA website. Owners can then schedule a time to have the update installed.

There are many (and cheap) micro-computers



Arduino





Best 52 single-board computers, 2019: [https://www.slant.co/topics/1629/~best-single-board-computers]

eCos = embedded configurable operating system

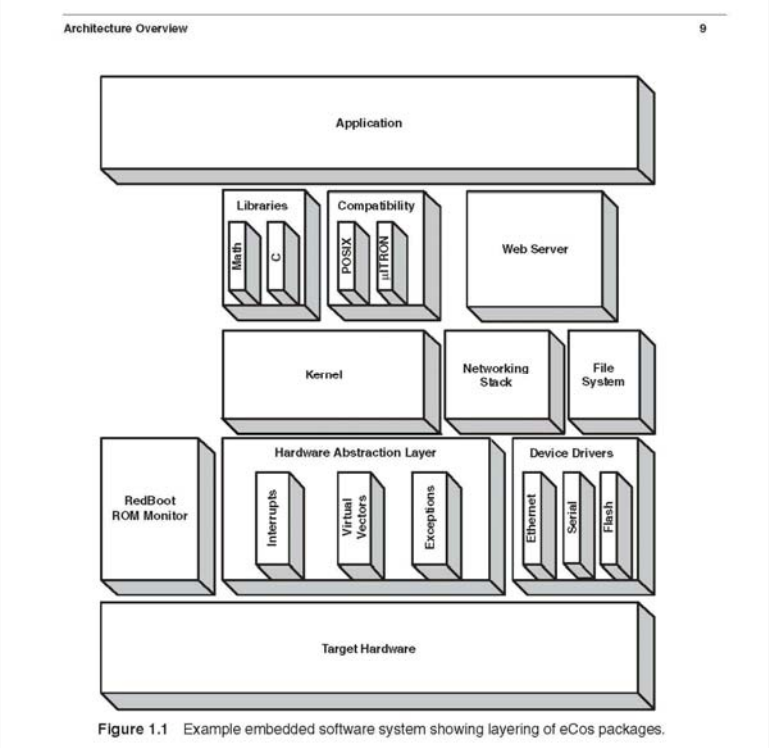
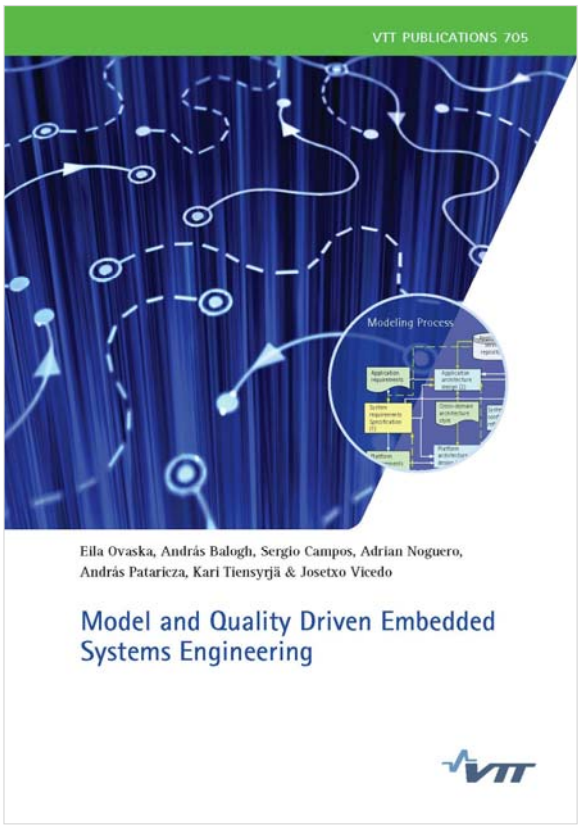


Figure 1.1 Example embedded software system showing layering of eCos packages.



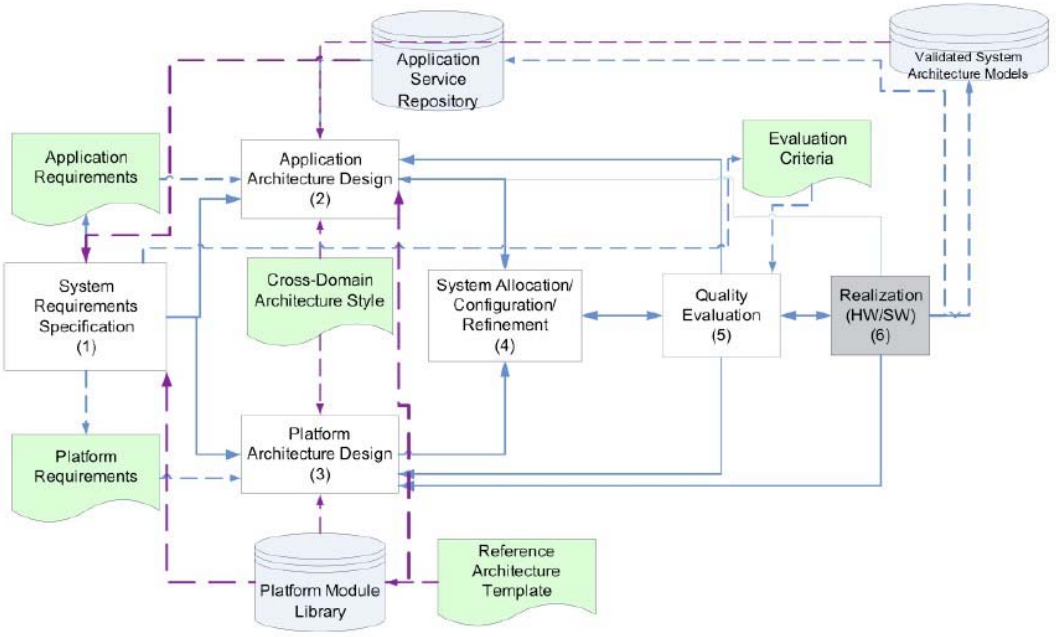


Figure 3. Overview of the process model.

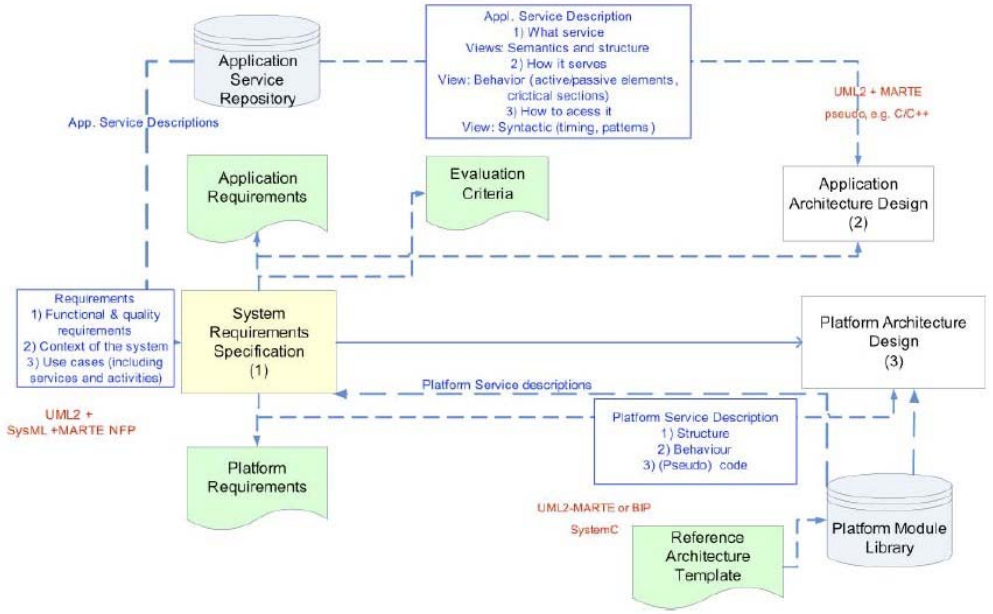


Figure 4. Overview of the System Requirements Specification phase.

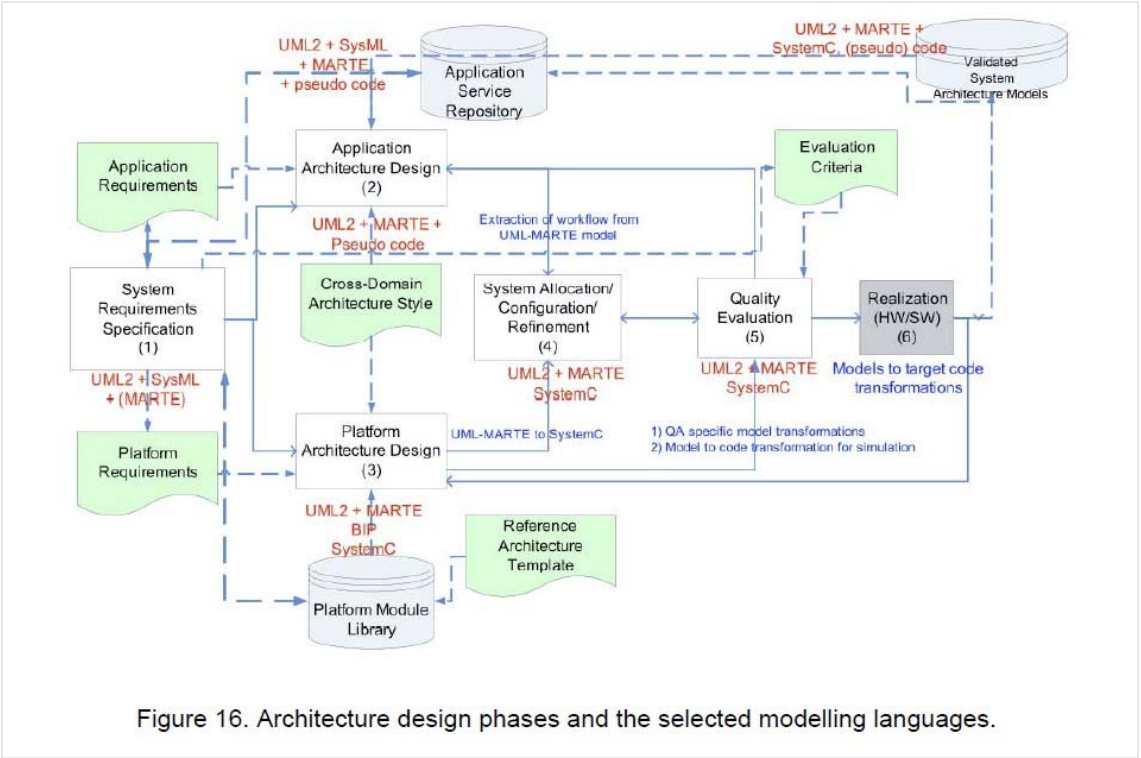


Figure 16. Architecture design phases and the selected modelling languages.

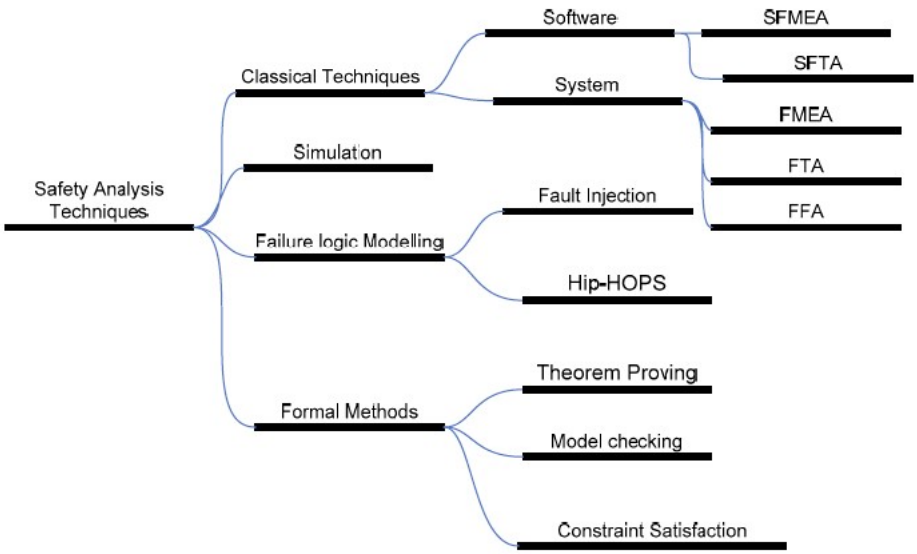


Figure 45. Taxonomy of safety analysis techniques.

Tampereen yliopisto
Tampere University

AI, artifical intelligence

Tampereen yliopisto
Tampere University

AI, artifical intelligence

AI = artifical intelligence (FI: tekoäly, apuäly, keinoäly)

AGI = artifical general intelligence

https://bdtechtalks.com/tag/demystifying-ai/

Tampereen yliopisto
Tampere University

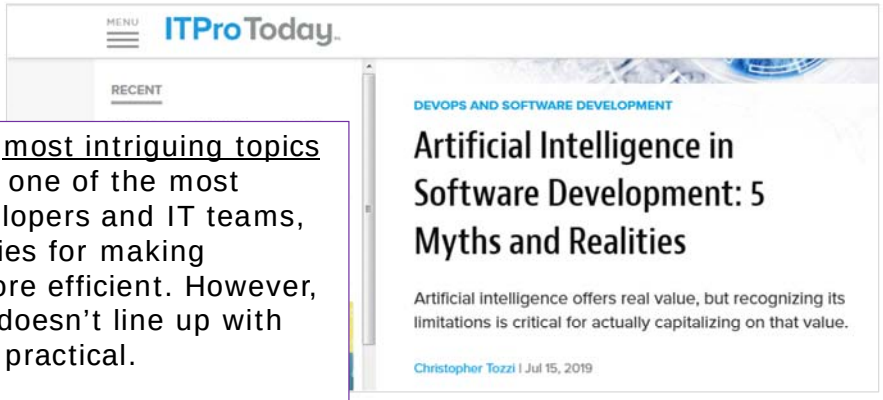
https://brainhub.eu/blog/software-developer-age-of-ai/

According to a team of researchers at the US Department of Energy's Oak Ridge National Laboratory, there's a high chance that **AI will replace software developers as early as 2040**.

"Programming trends suggest that software development will undergo a radical change in the future: the combination of machine learning, artificial intelligence, natural language processing, and code generation technologies will improve in such a way that machines, instead of humans, will write most of their own code by 2040," state the researchers.

Software developers are understandably worried. In fact, nearly 30 percent of the 550 software developers surveyed by Evans Data Corporation, a California-based market research firm that specializes in software development, believe that their development efforts will be replaced by artificial intelligence in the foreseeable future.

Tampereen yliopisto
Tampere University



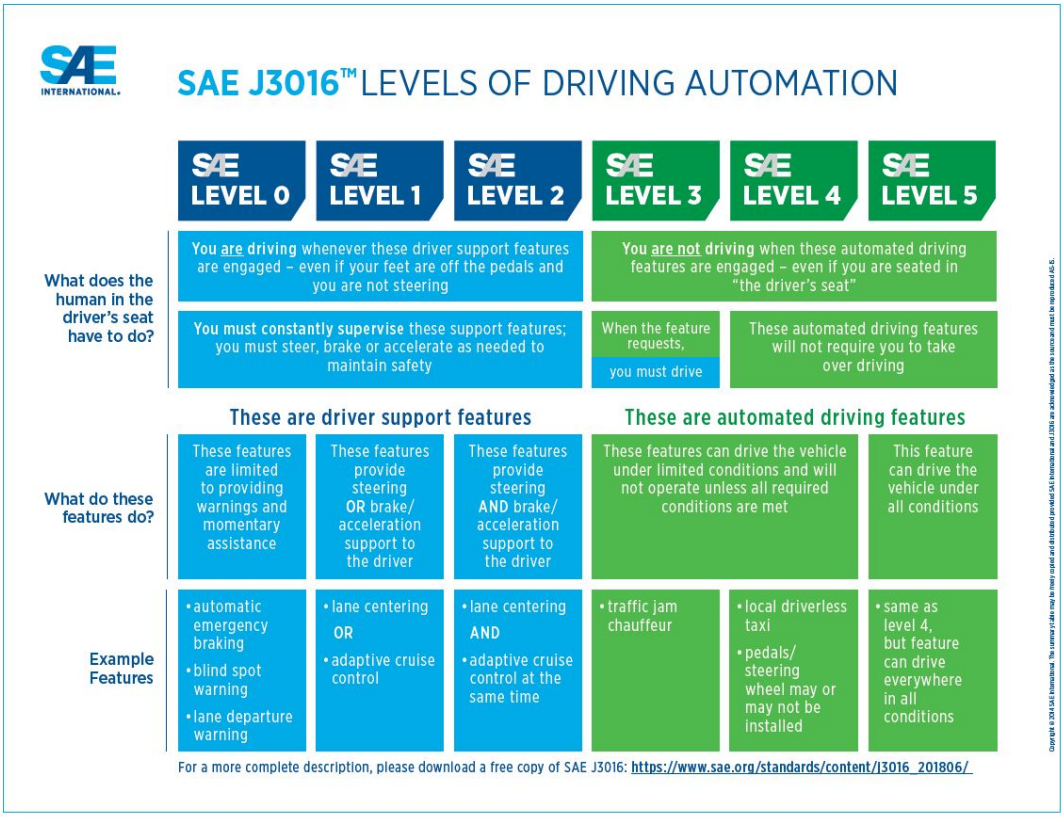
Artificial intelligence, or AI, is one of the most intriguing topics in software development today. It is also one of the most widely misunderstood. For software developers and IT teams, AI offers an array of tantalizing possibilities for making applications faster, more scalable and more efficient. However, in many cases, the hype surrounding AI doesn't line up with the reality of what is actually possible or practical.

1. **Artificial inteligence is a new technology.**
2. **AI is smarter than humans.**
3. **AI will lead to smaller IT teams.**
4. **AI software is "set and forget."**
5. **AI will destroy the world.**

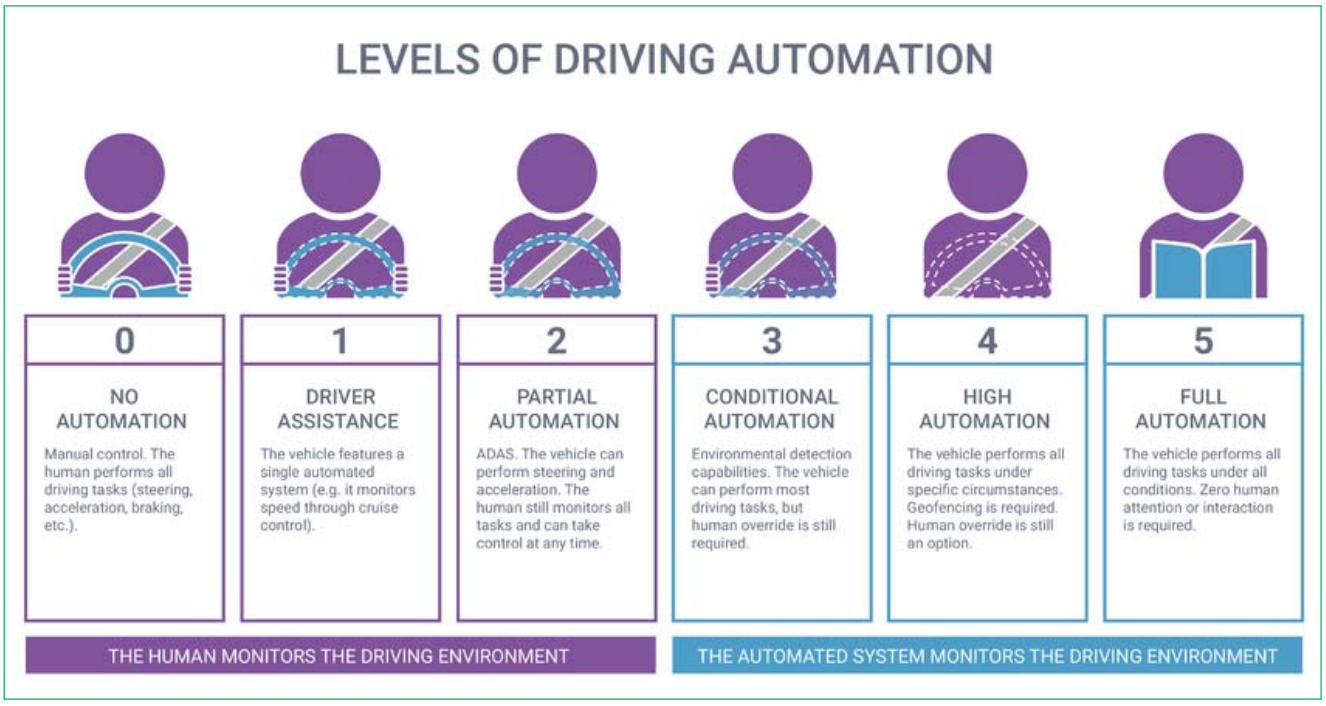
Tampereen yliopisto
Tampere University

SAE (Society of Automotive Engineers) **Six levels of driving automation, 2018**

[https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic]



Tampereen yliopisto
Tampere University



[https://www.synopsys.com/automotive/autonomous-driving-levels.html]

Uber-car accident 18.03.2018

A self-driving Uber Volvo XC90 SUV killed 49-year-old Elaine Herzberg as she walked her bicycle across a street in Tempe, Arizona, Sunday night, according to the Tempe Police Department.

Uber's self-driving car showed no signs of slowing before fatal crash, police say

The vehicle was traveling at 40 mph

ARIZONA · 1 day ago

Self-driving Uber car kills Arizona pedestrian, police say

Is this developer's (coder's) fault ??

Tampereen yliopisto
Tampere University

BUSINESS NEWS OCTOBER 10, 2018 / 6:12 AM / A YEAR AGO

Amazon scraps secret AI recruiting tool that showed bias against women

Jeffrey Dastin 8 MIN READ

SAN FRANCISCO (Reuters) - Amazon.com Inc's (AMZN.O) machine-learning specialists uncovered a big problem: their new recruiting engine did not like women.

The team had been building computer programs since 2014 to **review job applicants' resumes** with the aim of mechanizing the search for top talent, five people familiar with the effort told Reuters.

The company's experimental **hiring tool used artificial intelligence** to give job candidates scores ranging from one to five stars - much like shoppers rate products on Amazon, some of the people said.

[https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G]

Amazon's AI

"Everyone wanted this holy grail," one of the people said. "They literally wanted it to be an engine where I'm going to give you 100 resumes, it will spit out the top five, and we'll hire those."

But by 2015, the company realized its new **system was not rating candidates** for software developer jobs and other technical posts **in a gender-neutral way**.

That is because Amazon's computer models were trained to vet applicants by observing patterns in resumes submitted to the company over a 10-year period. Most came from men, a reflection of male dominance across the tech industry.

In effect, **Amazon's system taught itself that male candidates were preferable**. It penalized resumes that included the word "women's," as in "women's chess club captain." And it downgraded graduates of two all-women's colleges, according to people familiar with the matter. They did not specify the names of the schools.

Amazon edited the programs to make them neutral to these particular terms. But that was no guarantee that the machines would not devise other ways of sorting candidates that could prove discriminatory, the people said.

The Seattle company ultimately disbanded the team by the start of last year because executives lost hope for the project, according to the people, who spoke on condition of anonymity.

HYPE

Microsoft's disastrous Tay experiment shows the hidden dangers of AI

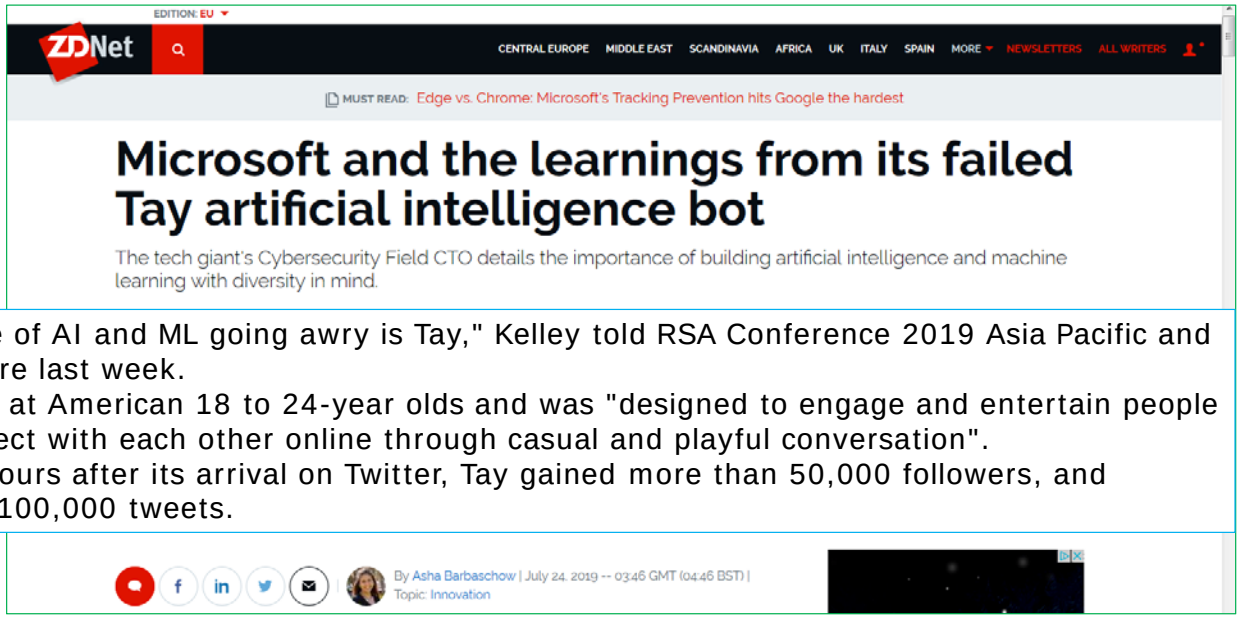
By John West • April 2, 2016

Microsoft's disastrous **chatbot Tay** was meant to be a clever experiment in artificial intelligence and machine learning. The bot would speak like millennials, **learning from the people it interacted with** on Twitter and the messaging apps Kik and GroupMe. But it took less than 24 hours for Tay's cheery greeting of "Humans are super cool!" to morph into the decidedly less bubbly "Hitler was right." Microsoft quickly took the bot offline for "some adjustments." Upon seeing what their code had wrought, one wonders if those Microsoft engineers had the words of J. Robert Oppenheimer ringing in their ears: "Now I am become death, the destroyer of worlds."

It's true that sometimes, humans were teaching Tay to hate. Daniel Victor at The New York Times writes: "Users commanded the bot to repeat their own statements, and the bot dutifully obliged."

[https://qz.com/653084/microsofts-disastrous-tay-experiment-shows-the-hidden-dangers-of-ai/]

Tampereen yliopisto
Tampere University



"A great example of AI and ML going awry is Tay," Kelley told RSA Conference 2019 Asia Pacific and Japan in Singapore last week.
Tay was targeted at American 18 to 24-year olds and was "designed to engage and entertain people where they connect with each other online through casual and playful conversation".
In less than 24 hours after its arrival on Twitter, Tay gained more than 50,000 followers, and produced nearly 100,000 tweets.

Tay started fairly sweet; it said hello and called humans cool. But Tay started interacting with other Twitter users and its machine learning (ML) architecture hoovered up all the interactions, good, bad, and awful.
Some of Tay's tweets were highly offensive. In **less than 16 hours Tay had turned into a brazen anti-Semite and was taken offline for re-tooling**.

Tampereen yliopisto
Tampere University

Tampereen yliopisto
Tampere University

Eleven key actions ushering Finland into the age of artificial intelligence

1: Enhance business competitiveness through the use of AI
2: Effectively utilise data in all sectors
3: Ensure AI can be adopted more quickly and easily
4: Ensure top-level expertise and attract top experts
5: Make bold decisions and investments
6: Build the world's best public services
7: Establish new models for collaboration
8: Make Finland a forerunner in the age of artificial intelligence
9: Prepare for artificial intelligence to change the nature of work
10: Steer AI development into a trust-based, human-centred direction
11: Prepare for security challenges.

Tampereen yliopisto
Tampere University

The following key measures should be introduced during the next 12 months (1/2):

Clarify the **rules of how data is used**, from the perspective of companies, society and users. Provide support for the use of data by means of legislation, agreements and self-regulation of industries.

Support the development of **significant testbeds** and international cooperation. Integrate the operations as part of the Finnish Digital Innovation Hub network.

Recognise the **business potential of different types of ecosystems and the B2B market** and develop solutions for using data in them.

Continue **AI accelerator style operations** based on the lessons learned and seek opportunities to expand the operations.

Ensure Finland's ability to **secure major strategic investments** in AI and RDI investments in competences.

On the basis of the experiences gained, create an **extensive provision of online courses for those in working life**, which would provide an opportunity for the adult population to supplement and renew their competences.

[Leading the way into the era of artificial intelligence, 2019]

The following key measures should be introduced during the next 12 months (2/2):

Explore whether **every Finn in working age could be provided with a learning voucher or account**, which would create a well-functioning adult education market in Finland.

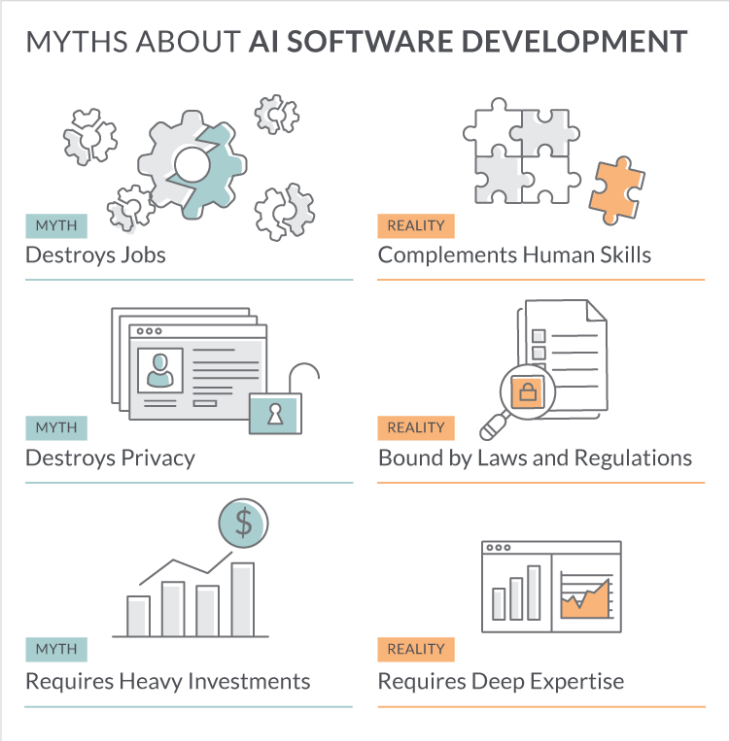
Ensure human-centric **introduction of artificial intelligence and the implementation of ethical principles in the public sector** through the AuroraAI project.

Encourage companies and public-sector actors to introduce **ethical self-regulation** and to share best practices.

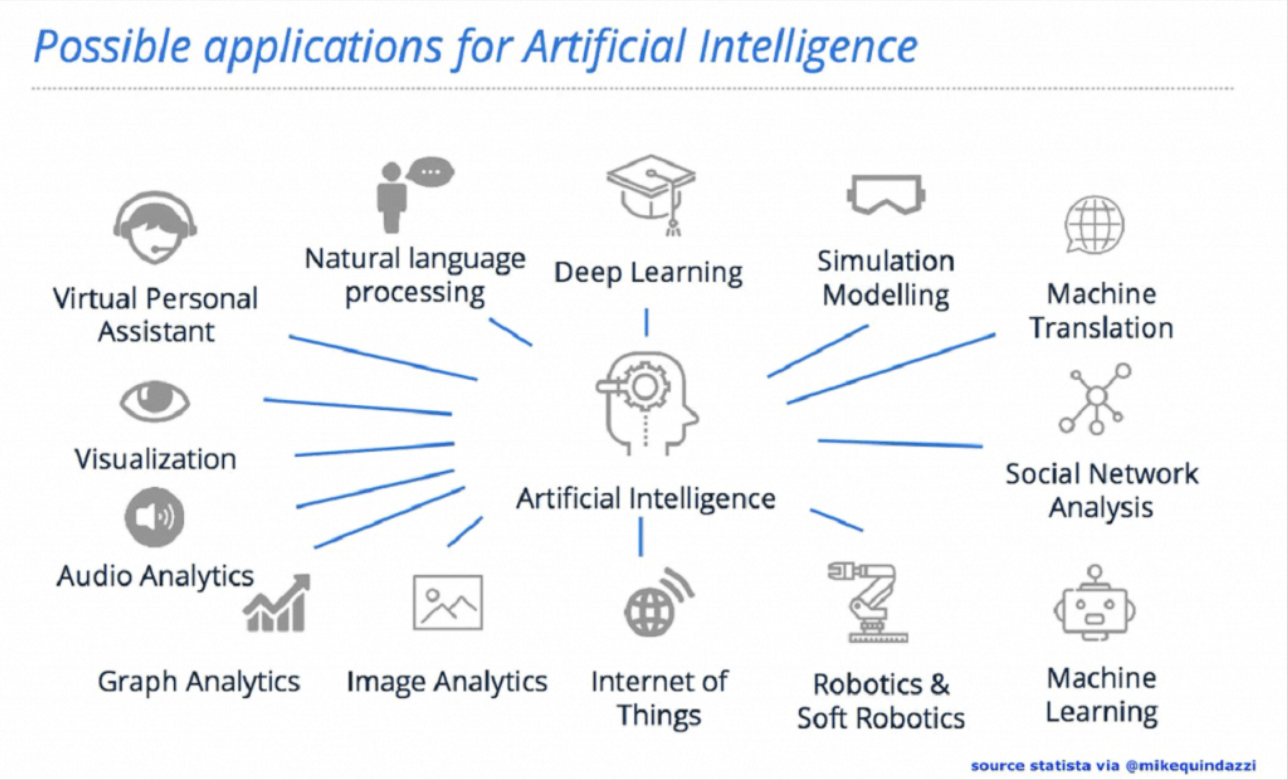
Introduce the digital economy, founded on artificial intelligence, data and platform economy, as one of the key themes of **Finland's EU Presidency**.

Monitor how the implementation of the Artificial Intelligence Programme's objectives is advancing. The responsibility for the monitoring should belong to a **monitoring group with representatives from both the private and public sectors** or a broader cooperation forum promoting the digitalisation of business.

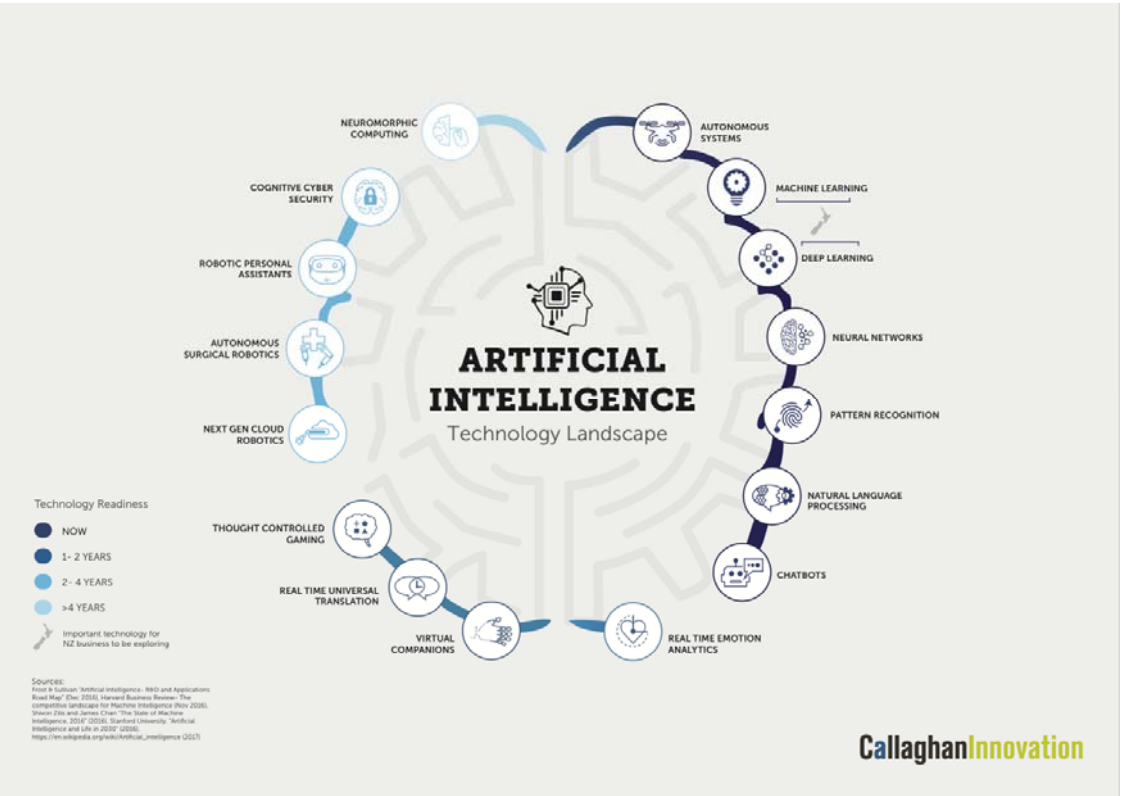
[Leading the way into the era of artificial intelligence, 2019]



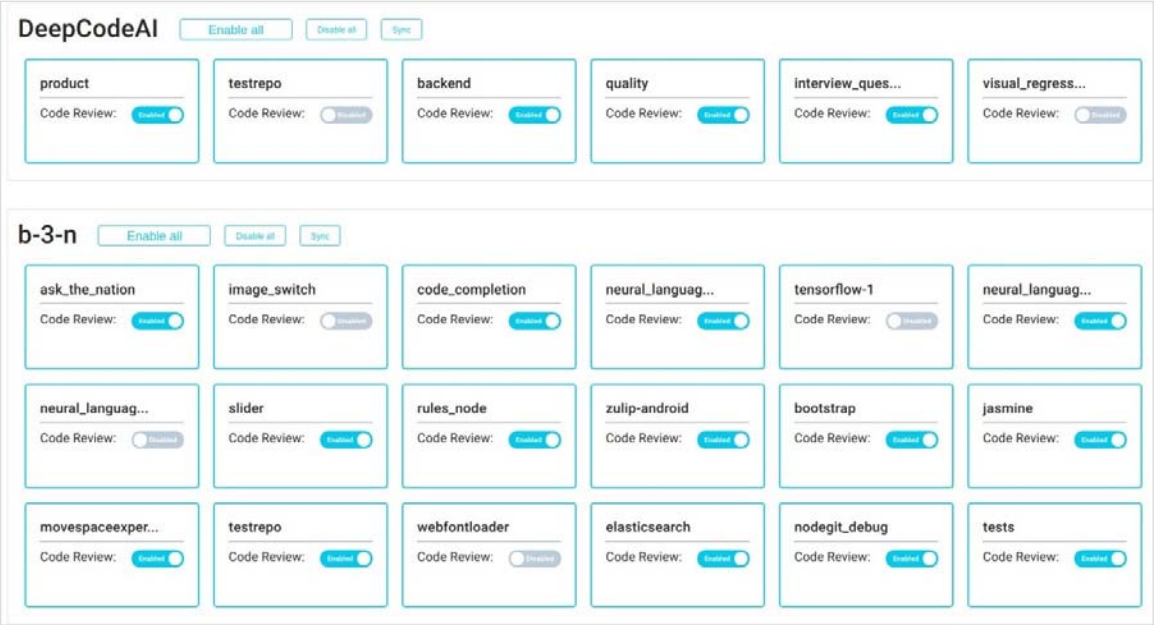
[https://www.experfy.com/blog/ai-software-development-dispelling-the-most-common-myths]



Some AI application areas



How AI Powered Tools Are Bringing Revolution to Software Development? 12 April 2019



5 ways artificial intelligence is upgrading software engineering

By Melisha Dsouza- September 2, 2018 - 6:00 pm

47% of digitally mature organizations, or those that have advanced digital practices, **said they have a defined AI strategy** (Source: Adobe). It is estimated that AI-enabled tools alone will generate $2.9 trillion in business value by 2021. 80% of enterprises are smartly investing in AI. The stats speak for themselves. AI clearly follows the motto “go big or go home”.

1 Software design

2 Software testing

3 GUI testing

4 Using Artificial Intelligence in Strategic Decision-Making

5 Automatic Code generation / Intelligent Programming Assistants.

Software engineering has seen massive transformation over the past few years. AI and software intelligence tools **aim to make software development easier and more reliable**. According to a Forrester Research report on AI’s impact on software development, automated testing and bug detection tools use AI the most to improve software development.

Mariya Yao : **6 Ways AI Transforms How We Develop Software** , Apr 18, 2018 (1/2)

1. Rapid Prototyping. Turning business requirements into technology products typically requires months if not years of planning, but machine learning is shortening this process by enabling less technical domain experts to develop technologies using either natural language or visual interfaces.

2. Intelligent Programming Assistants. Developers spend the vast majority of their time reading documentation and debugging code. Smart programming assistants can reduce this time by offering just-in-time support and recommendations, such as relevant document, best practices, and code examples. Examples of such assistants include Kite for Python and Codota for Java.

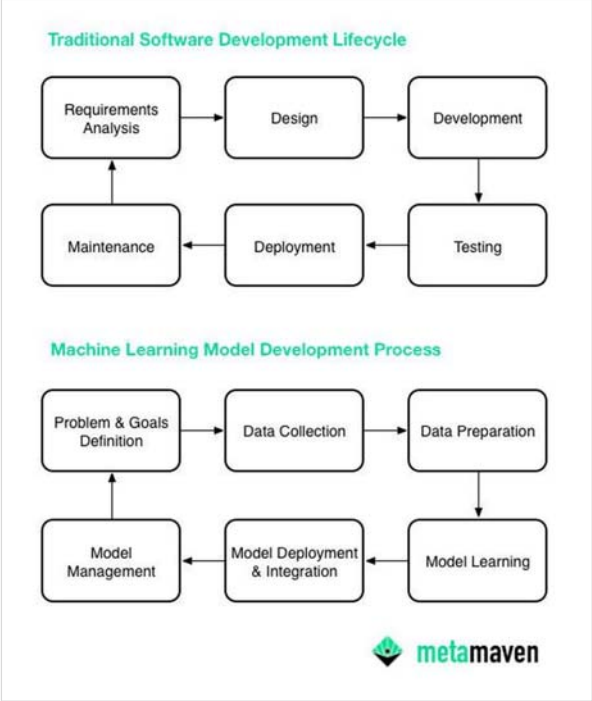
3. Automatic Analytics & Error Handling. Programming assistants can also learn from past experience to identify common errors and flag them automatically during the development phase. Once a technology has been deployed, machine learning can also be used to analyze system logs to quickly and even proactively flag errors. In the future, it would also be possible to enable software to change dynamically in response to errors without human intervention.

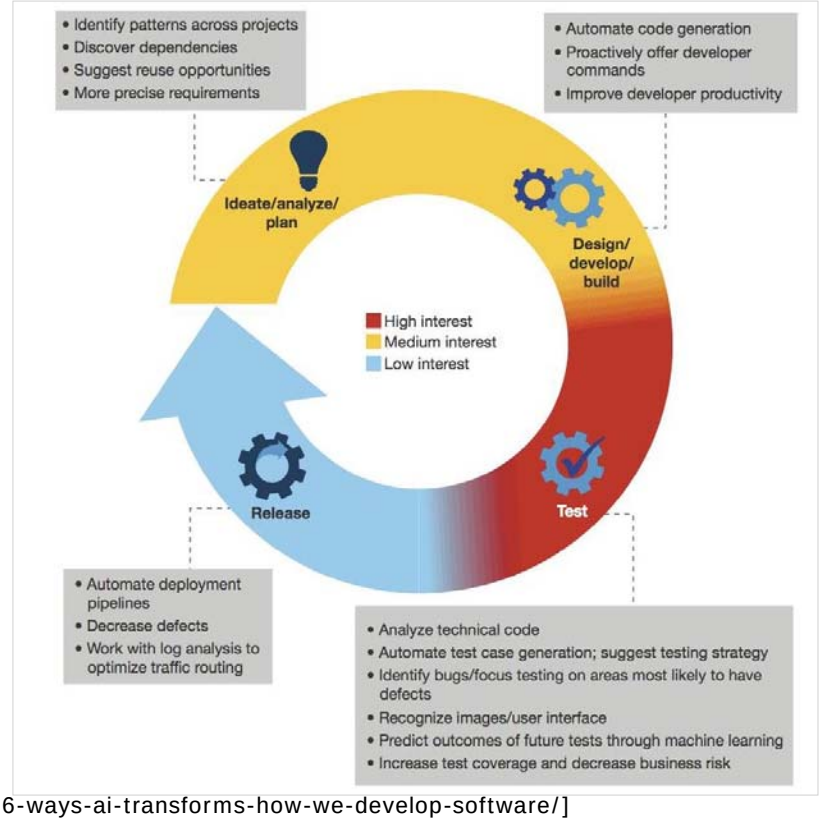
Mariya Yao : **6 Ways AI Transforms How We Develop Software** , Apr 18, 2018 (2/2)

4. Automatic Code Refactoring. Clean code is critical for team collaboration and long-term maintenance. As enterprises upgrade their technologies, large-scale refactoring are unavoidable and often painful necessities. Machine learning can be used to analyze code and automatically optimize it for interpretability and performance.

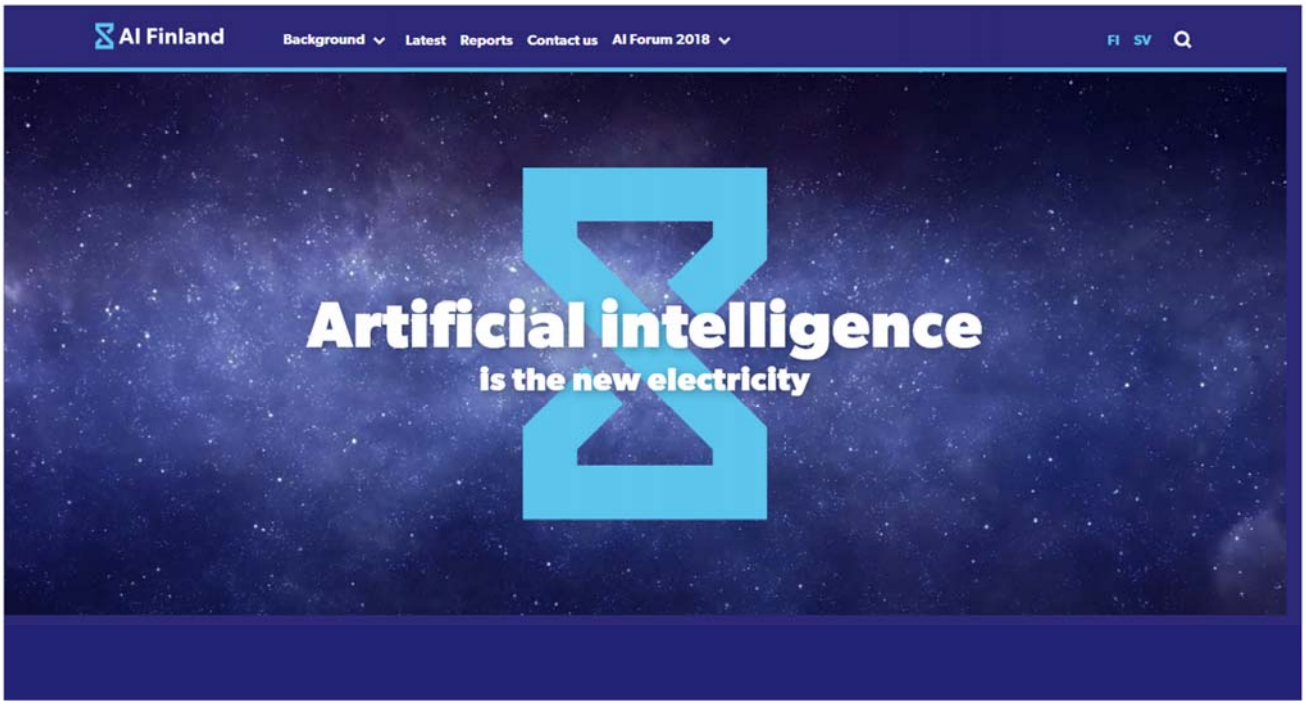
5. Precise Estimates. Software development notoriously goes over budget and over timelines. Reliable estimates require deep expertise, understanding of context, and familiarity with the implementation team. Machine learning can train on data from past projects - such as user stories, feature definitions, estimates, and actuals - to predict effort and budget more accurately.

6. Strategic Decision-Making. A significant portion of time is spent debating which products and features to prioritize and which to cut. An AI solution trained on both past development projects and business factors can assess the performance of existing applications and help both business leaders and engineering teams identify efforts that would maximize impact and minimize risk.





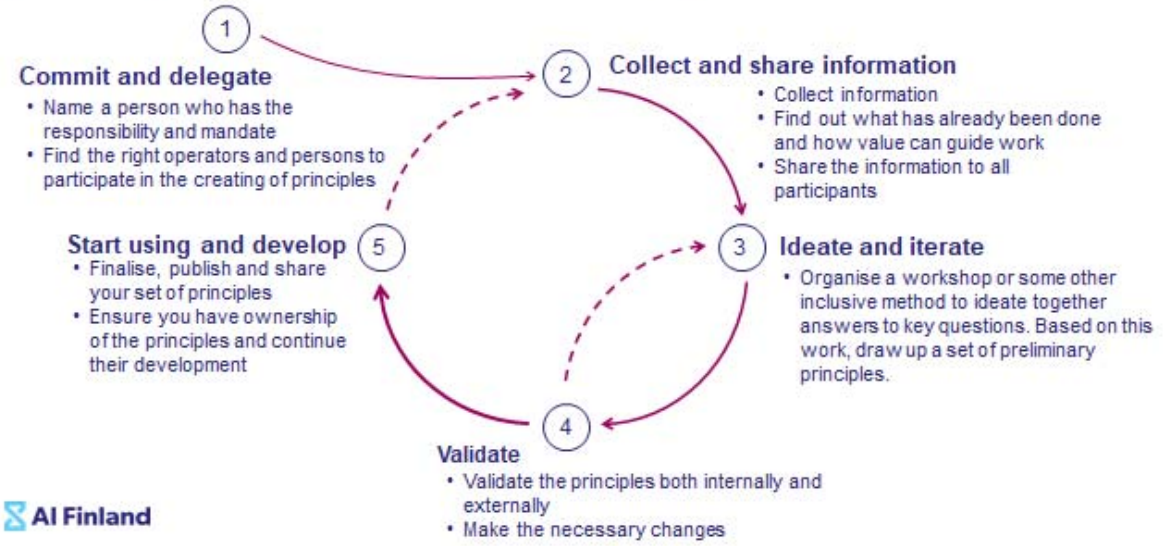
[https://www.forbes.com/sites/mariyayao/2018/04/18/6-ways-ai-transforms-how-we-develop-software/]



[https://www.tekoalyaika.fi/en/]

Tampereen yliopisto
Tampere University

Five steps to defining the ethical principles of artificial intelligence



AI Finland

Tampereen yliopisto
Tampere University

Manifesto for Data Practices

datapractices.org

1) Use data to improve life for our users, customers, organizations, and communities.
2) Create reproducible and extensible work.
3) Build teams with diverse ideas, backgrounds, and strengths.
4) Prioritize the continuous collection and availability of discussions and metadata.
5) Clearly identify the questions and objectives that drive each project and use to guide both planning and refinement.
6) Be open to changing our methods and conclusions in response to new knowledge.
7) Recognize and mitigate bias in ourselves and in the data we use.
8) Present our work in ways that empower others to make better-informed decisions.
9) Consider carefully the ethical implications of choices we make when using data, and the impacts of our work on individuals and society.
10) Respect and invite fair criticism while promoting the identification and open discussion of errors, risks, and unintended consequences of our work.
11) Protect the privacy and security of individuals represented in our data.
12) Help others to understand the most useful and appropriate applications of data to solve real-world problems.

AI Finland

https://datapractices.org/manifesto/

#tekoälyaika | #aiera

Tampereen yliopisto
Tampere University

Statement on AI, Robotics and Autonomous Systems

European Group on Ethics in Science and New Tech (EGE)

1) **Human dignity**:
   Limits to classifications & awareness whether we are interacting with a machine or human
2) **Autonomy**:
   Human ability to choose whether to delegate decisions and actions to AI or not
3) **Responsibility:**
   AI should be developed in ways serving social good as determined by democratic processes
4) **Justice, equity, and solidarity**:
   No discriminatory bias in datasets & equal access to AI tech & fair distribution of benefits
5) **Democracy:**
   Key decisions on regulation and application results of democratic and public debate
6) **Rule of law and accountability**:
   Protection agains risks stemming from AI that infringes human rights eg safety and privacy
7) **Security, safety, bodily and mental integrity**:
   All safety dimensions taken into account in development and tested before release
8) **Data protection and privacy:**
   Also limit for tech influencing personal opinions
9) **Sustainability:** Priority for environmental protection

AI Finland

http://ec.europa.eu/research/ege/pdf/ege_ai_statement_2018.pdf

#tekoälyaika | #aiera

Tampereen yliopisto
Tampere University

Ethical guidelines for the use of AI

OP Group

1) **People-oriented approach**

   We deploy data and AI responsibly to promote the wellbeing of our customers. We define clearly the goals of our AI work and refine them when necessary to respond to changes in data, technical possibilities and our work environment.

2) **Openness and transparency**

   We operate openly in relation to our customers, partners and stakeholders and ensure the transparency of our AI applications and their evaluation. We are open about the ways we use AI, and we subject our work to review.

3) **Impact assessment**

   We examine carefully how our choices affect our customers and their environments and strive always to make responsible choices when we apply AI.

4) **Ownership**

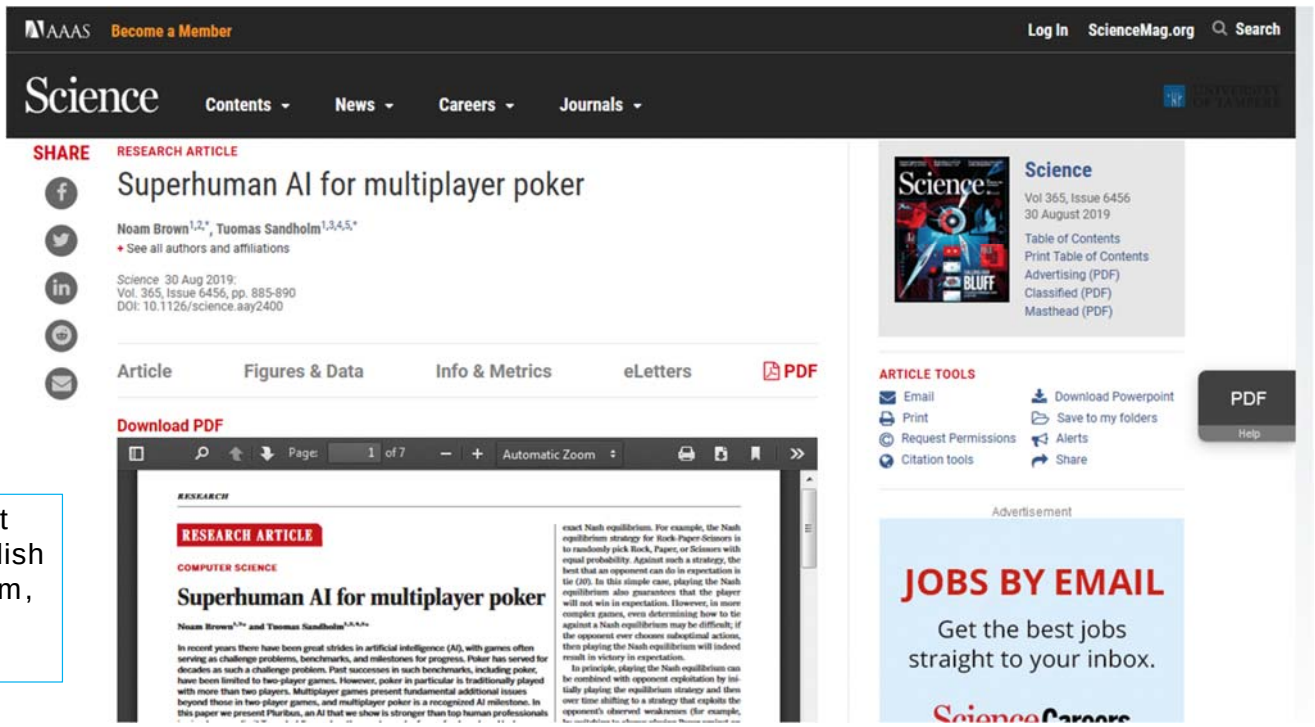
   We assign owners to all the principles guiding our work and all the algorithms we develop. We ensure that the AI we use is ethical throughout its life cycle.

5) **Privacy protection**

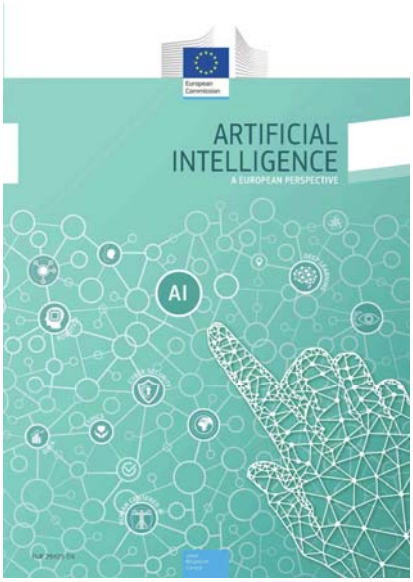
   We safeguard the protection of privacy and personal data in the data we use in accordance with our data protection policies.

AI Finland

#tekoälyaika | #aiera



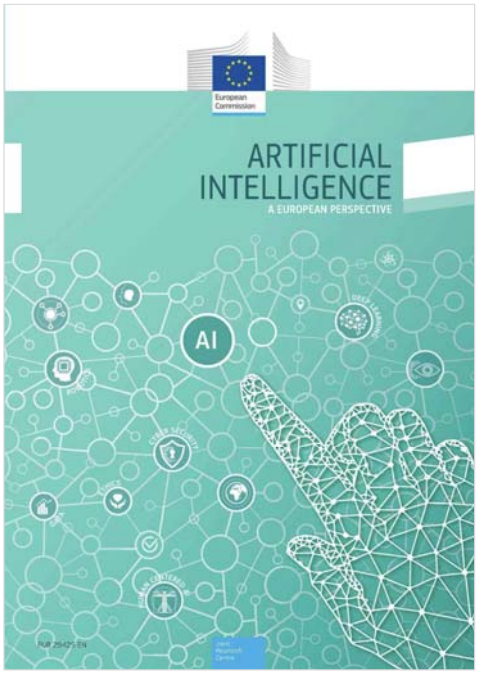
They did not dare to publish the algorithm, may set up chaos...



Data is the lifeline of AI.

Europe is data rich but the economics and legal framework create complex challenges.

Opening access to data and building interactions among participants is key to succeeding.



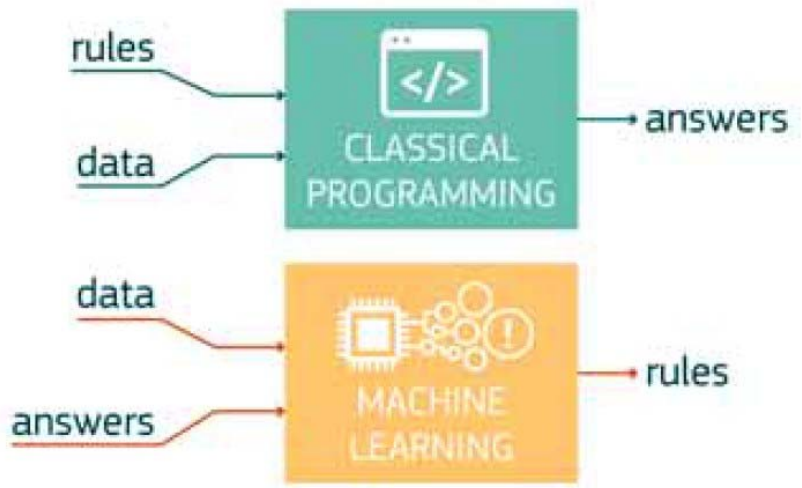


Figure 1. Paradigm shift in ML[2]
Source: after Chollet, 2017

BOX 4. Some examples of AI developments in the USA

Google is an AI world leader. It researches and develops AI products and services, has developed its own AI chip, the Tensor Processing Unit (TPU), and created TensorFlow, one of the most widely used open source AI/ML libraries. DL algorithms are already a considerable part of several Google products including recommendation systems, Android, Gmail, Maps, and translation services. The number of DL directories Google uses has gone from zero in 2012 to 4 000 in 2016.[14] In May 2018, Google announced Google duplex, an AI system for accomplishing real-world tasks over the phone. Google, and other large operators, also use AI extensively to manage their data centres and save energy.[15]

Facebook is the biggest social media company in the world. It has established a Facebook AI research (FAIR) team which is one of the most advanced, with several labs around the world. Facebook supports the open source AI libraries PyTorch and Caffe2 that compete with Google TensorFlow. The company has also released plans to develop its own AI chips.

Amazon uses AI extensively in its recommendation systems and logistics. It also offers both consumer and business-oriented AI products and services. Amazon Echo brings AI into the home through the intelligent voice server, Alexa. For business, Amazon AI empowers fake reviews detection, chatbots, product recommendations, big data management, etc. Amazon Web Services are probably the biggest cloud system in the world providing infrastructure as a service to over 1 million users.[16]

Microsoft has also been investing heavily in AI and infrastructure as a service. Microsoft Azure is one of the three biggest cloud (and AI) providers together with Google Cloud and Amazon web services. Cortana is the Microsoft virtual assistant that competes with Amazon Alexa, Apple Siri, Google Duplex and others.

Apple is the most valuable company in the world worth over US$ 1 trillion. Apple's main AI divisions are Siri team, and the Core ML team. Siri's team focus is on NLP and computer vision, both of which are necessary to power voice assistant features and new, more cutting-edge technology such as augmented reality apps that rely on object recognition. Core ML is the ML Application Programming Interface (API) that Apple launched last year to help AI tasks and AI-focused apps and services from third-party developers run more efficiently on Apple devices.

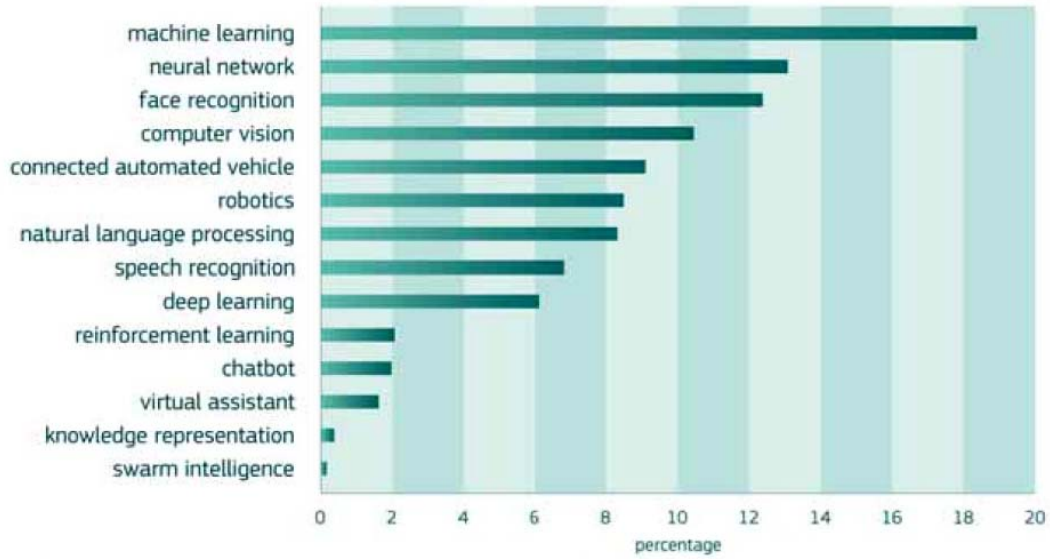
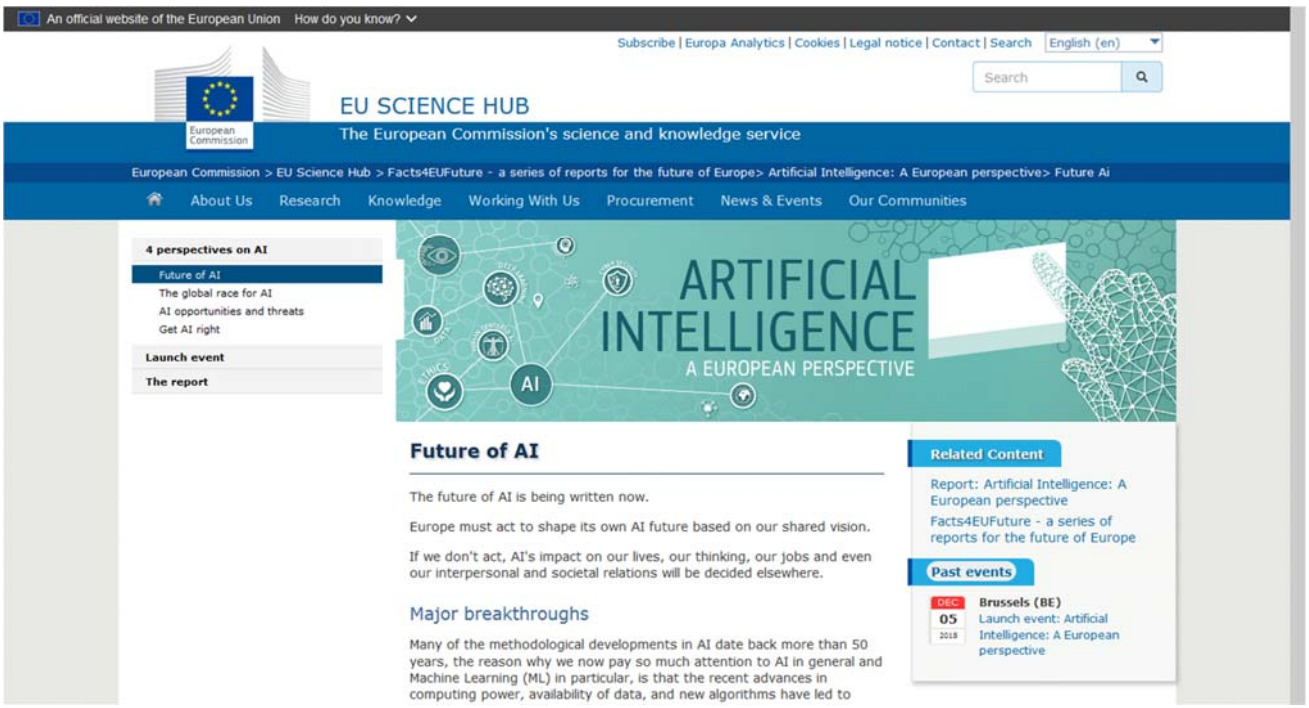
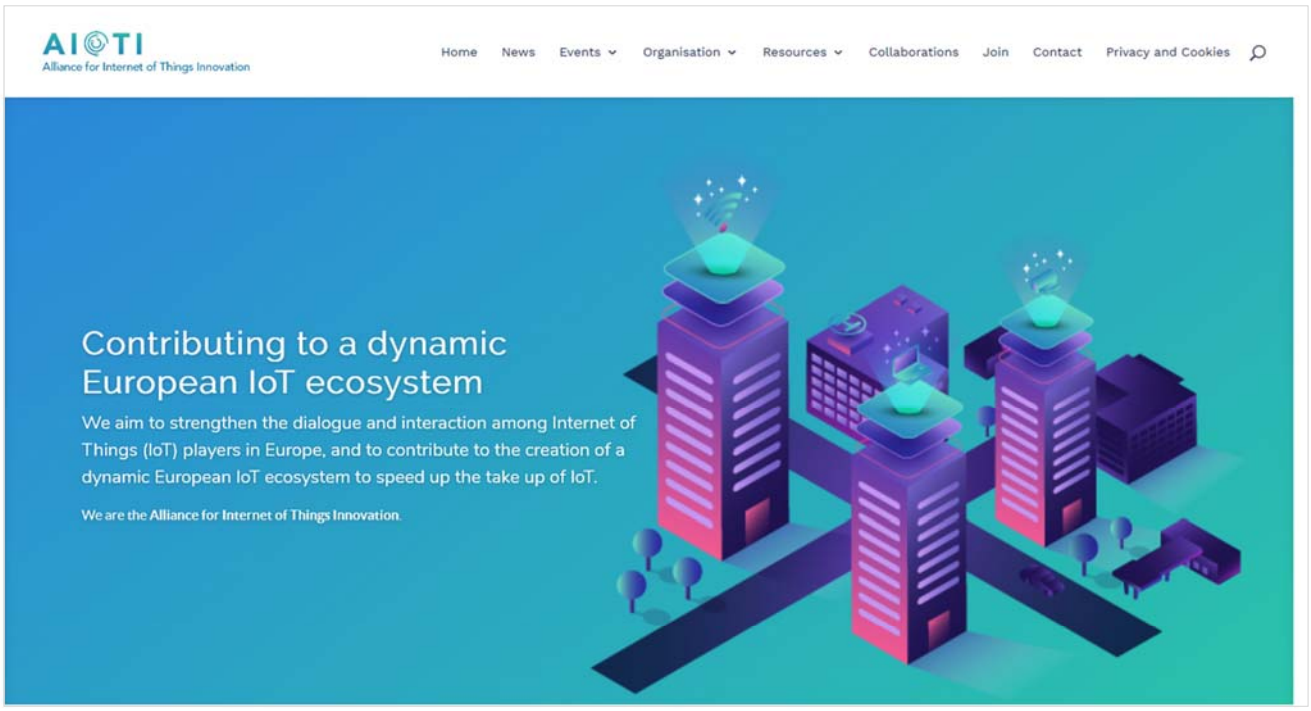


Figure 2. Percentage of AI players detected by the most representative keywords (2000-18)
Note: The conceptual proximity of some of the keywords, and the overlap between techniques and applications, may result in the same player been detected more than once.





[https://aioti.eu/]

BOX 9. Summary data strategy from the Mission Villani Report (2018)

1. Encourage companies to pool and share their data

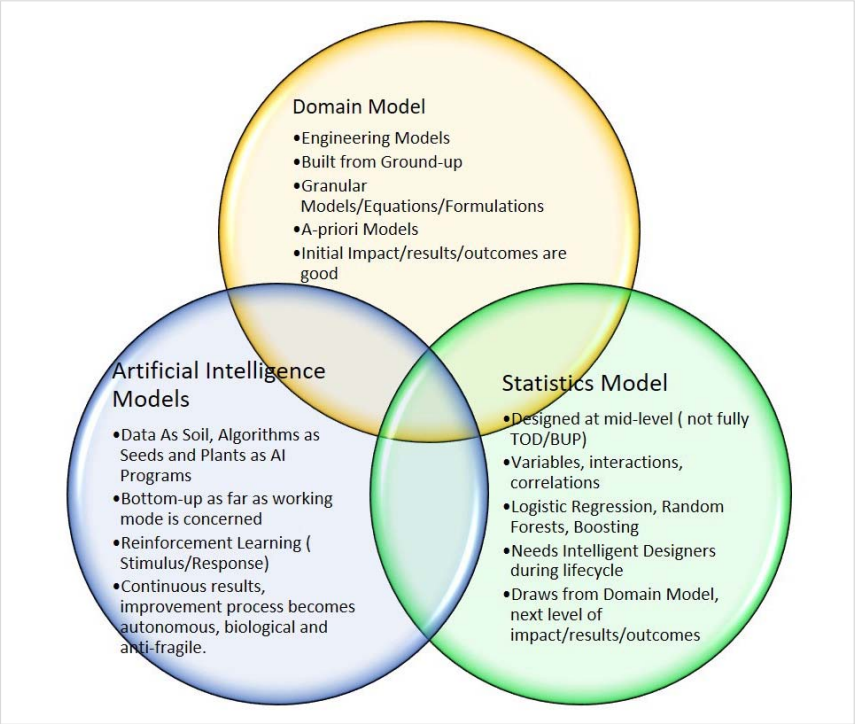
The government must encourage the creation of data commons and support an alternative data production and governance model based on reciprocity, cooperation and sharing. The goal is to boost data sharing between actors in the same sector. The government must also encourage data sharing between private actors, and assist businesses in this respect. It must organise for certain data held by private entities to be released on a case-by-case basis, and support data- and text-mining practices without delay.

2. Create data that is in the public interest

Most of the actors heard by the mission were in favour of progressively opening up access to some data sets on a case-by-case and sector-specific basis for public interest reasons. This could be in one of two ways: by making the data accessible only to the government, or by making the data more widely available, for example, to other economic actors.

3. Support the right to data portability

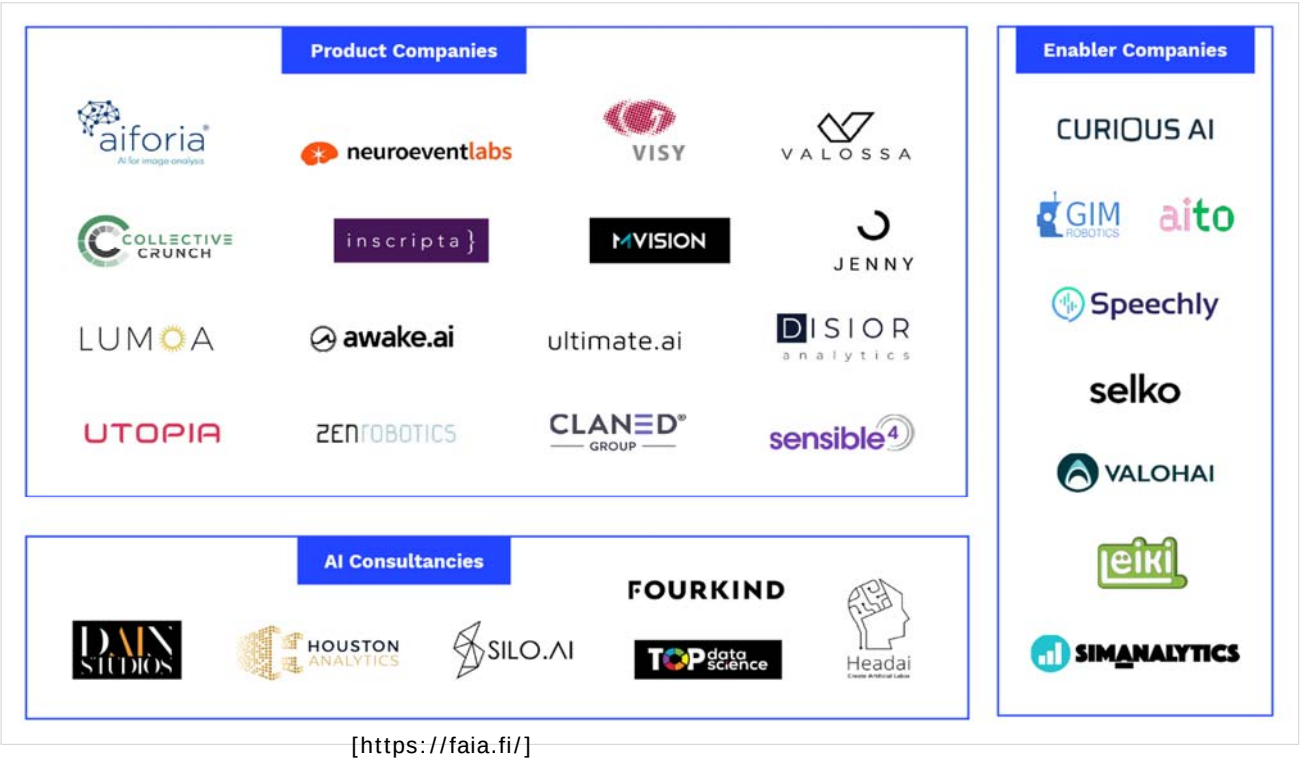
The right to data portability is one of the most important innovations in recent French and European texts. It will give any individual the ability to migrate from one service ecosystem to another without losing their data history. This right could be extended to all citizen-centred artificial intelligence applications. In this case, it would involve making personal data available to government authorities or researchers.



[https://medium.com/@akashmavle/domain-models-statistical-models-and-ai-ml-models-6314f03497d5]

[https://faia.fi/]

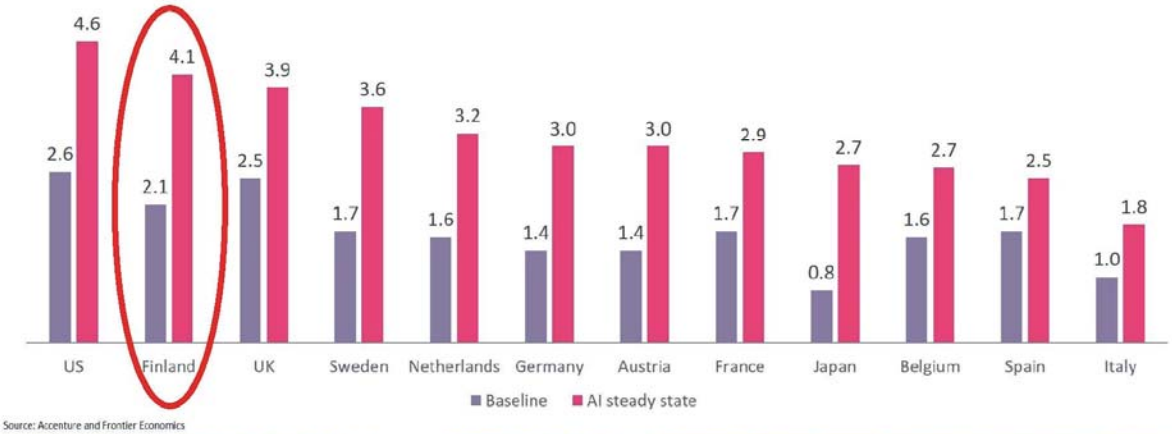
Finland's AI landscape



[https://faia.fi/]

Finland with and without Artificial Intelligence

Annual growth rates by 2035 of gross value added (a close approximation of GDP), comparing baseline growth by 2035 to an artificial intelligence scenario where AI has been absorbed into the economy



Source: Accenture and Frontier Economics

BUSINESS FINLAND Tekes Ministry of Economic Affairs and Employment of Finland

[Pekka Sivonen: FINLAND AND AI INTHE ERA OF PLATFORM ECONOMY, 2018]

Tampereen yliopisto
Tampere University

FINNISH APPROACH ON AI

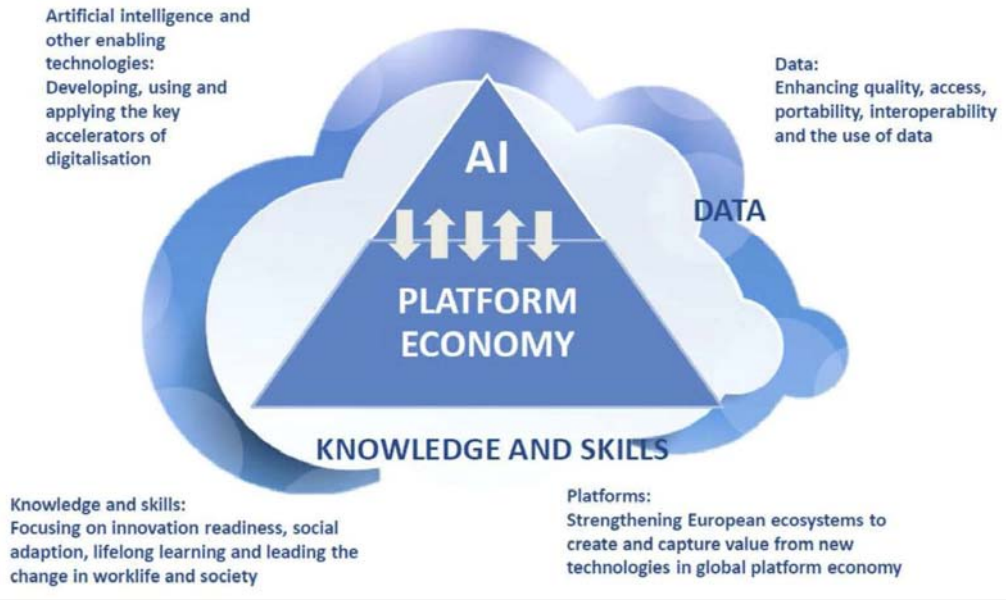
1. Increasing competitiveness of Finnish Businesses
2. Leveraging data extensively in all sectors
3. Speeding and easing up use of AI
4. Securing top-talent and attracting new talent (Top-Institute)
5. Courageous selections and investments
6. Building best Public Services worldwide with the help of AI
7. **Creating new models for global collaboration**
8. Positioning Finland as a role-model for the Age of AI

BUSINESS FINLAND Tekes FINNISH GOVERNMENT Ministry of Economic Affairs and Employment of Finland

[Pekka Sivonen: FINLAND AND AI INTHE ERA OF PLATFORM ECONOMY, 2018]

Tampereen yliopisto
Tampere University

Towards New Digital Europe



[Pekka Sivonen: FINLAND AND AI INTHE ERA OF PLATFORM ECONOMY, 2018]

Tampereen yliopisto
Tampere University

https://hackernoon.com/building-ai-software-data-driven-vs-model-driven-ai-and-why-we-need-an-ai-specific-software-640f74aaf78f , 2018

Data-driven AI

The data-driven way focusses on building a system that can identify what is the right answer based on having "seen" a large number of examples of question / answer pairs and "training" it to get to the right answer.

There are lots of different ways of doing this, with perhaps the most popular being using neural network algorithms in their various forms.

The necessary ingredients for this approach are an appropriately large dataset that, crucially, is also correctly labelled.

Model-driven AI

Model-driven AI (or **symbolic AI**), instead, attempts to capture knowledge and derive decisions through explicit representation and rules. In a model-driven world, a cat would be explicitly represented as a four-legged animal, with two eyes, a nose and a mouth that is furry (except when not) and that is relatively small (except when not), etc. A model-based system would look at an image, deconstruct it into lines and shapes and colours and the compare against the set of rules we've supplied about how lines and shapes and colours combine in the world to give us different animals.

You can immediately see why this is not a very good way of building a system to recognise a cat. There are so many different rules and exceptions to those rules that we can't capture all of them. More fundamentally, perhaps, we as humans don't actually know how we do it.

Tampereen yliopisto
Tampere University

cyber security

Tampereen yliopisto
Tampere University

”basic cybersecurity is OK”

First in Europe in Finland; Traficom mark for secure smart device

Kotimaa Julkaistu 26.11.2019 11:38

Suomi otti käyttöön Tietoturvamerkin älylaitteiden turvallisuuden varmistamiseen

Tietoturvamerkki takaa kuluttajalle, että laitteen tietoturvan perusominaisuudet ovat kunnossa. LEHTIKUVA / ANTTI AIMO-KOIVISTO

Suomi on aloittanut älylaitteiden turvallisuuden varmistamisen ensimmäisenä Euroopassa. Uusi Tietoturvamerkki auttaa kuluttajia tekemään turvallisempia älylaitehankintoja kotiin, tiedottaa Liikenne- ja viestintävirasto Traficom.

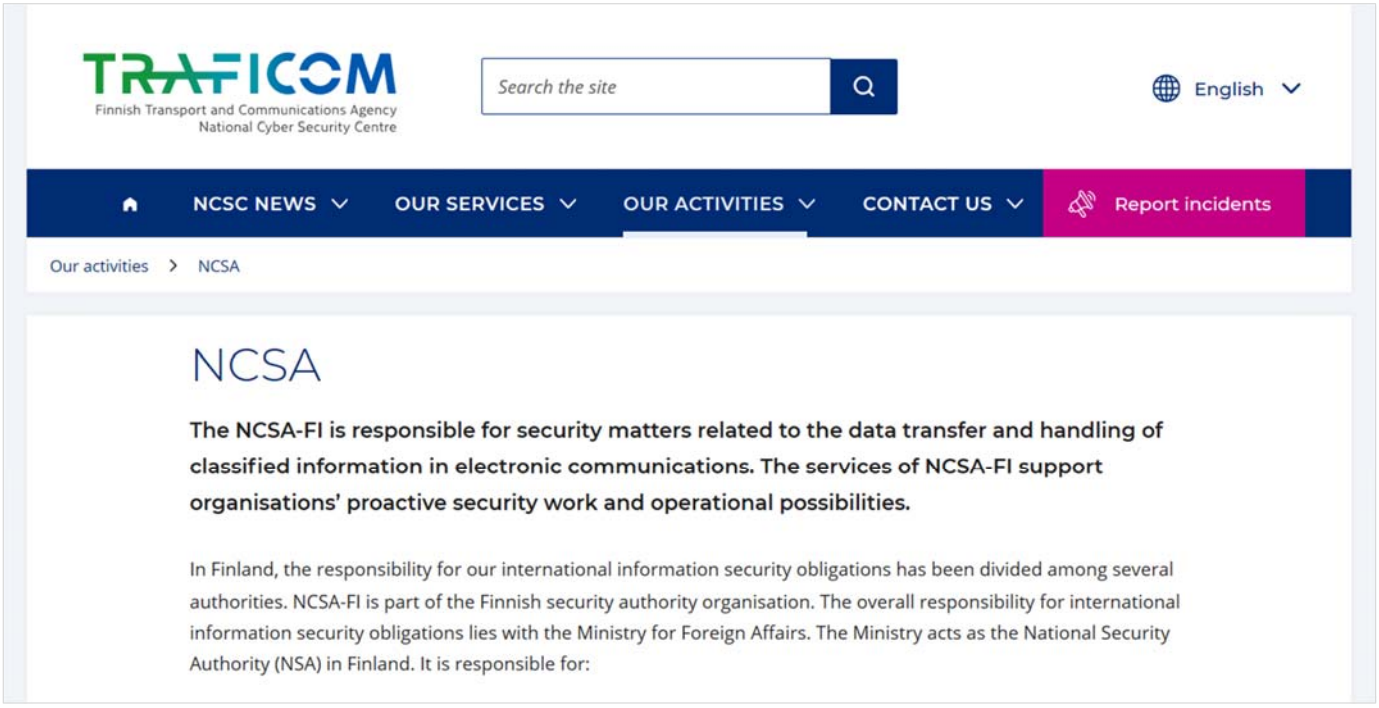
Traficom julkaisi Tietoturvamerkin tiistaina. Merkki takaa kuluttajalle, että laitteen tietoturvan perusominaisuudet ovat kunnossa.

Fiksumpia älylaite-hankintoja.

TRAFICOM
Liikenne- ja viestintävirasto
Kyberturvallisuuskeskus

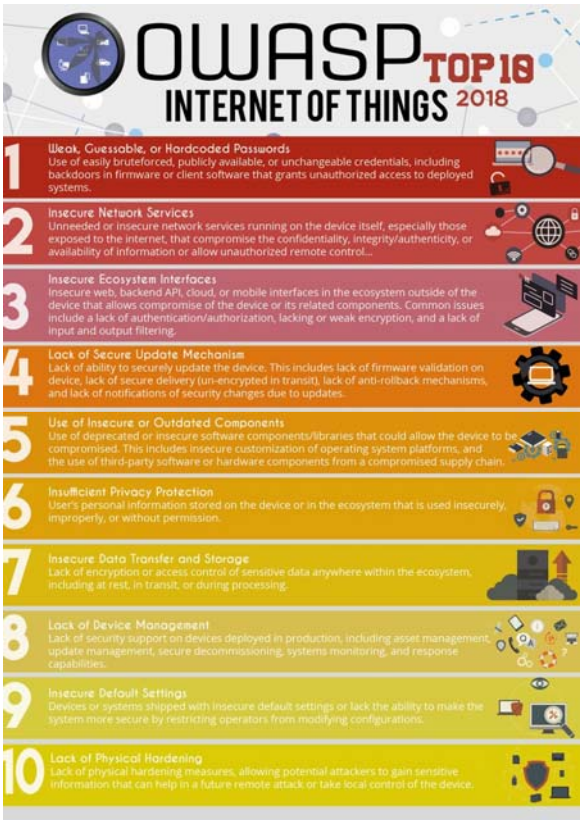
Tampereen yliopisto
Tampere University

www.kyberturvallisuuskeskus.fi/en/



Tampereen yliopisto
Tampere University

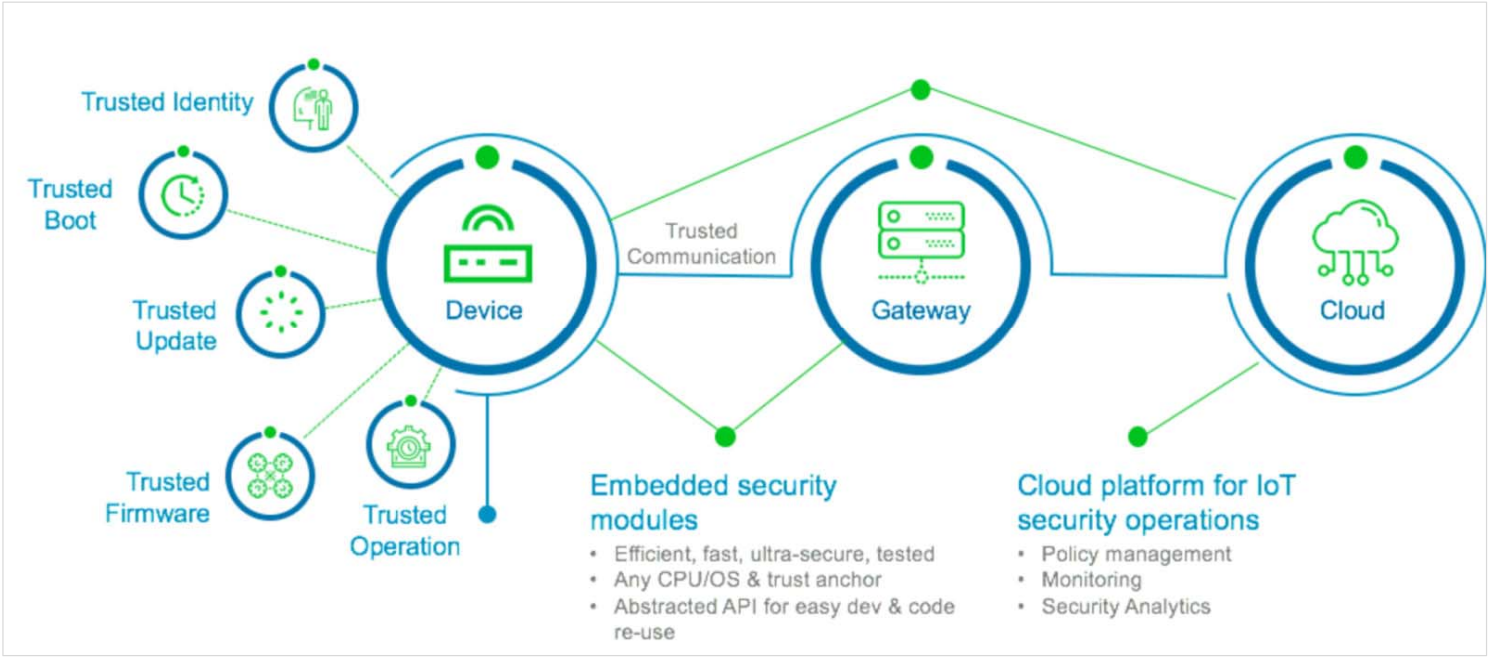
The Open Web Application Security Project (OWASP)



[www.owasp.org/]

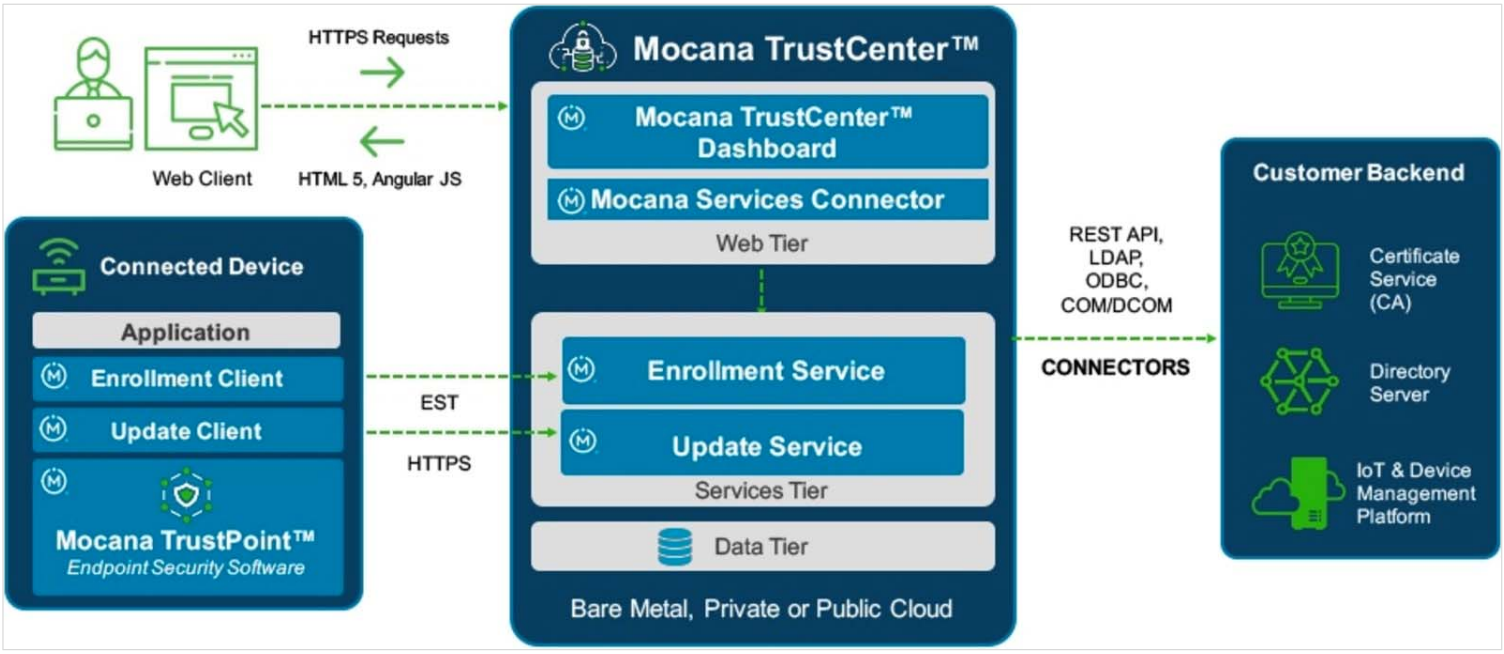
Tampereen yliopisto
Tampere University

IoT securtiy



[https://www.engineering.com/IOT/ArticleID/15354/End-to-End-IoT-Security.aspx]

Example of IoT security architecture



[https://www.engineering.com/IOT/ArticleID/16867/Mocana-TrustCenter-Joins-TrustPoint-in-Securing-IoT-Devices.aspx?e_src=relart]

The military has a phrase that you can get all other ”professionals” just by promoting, but not pilots and cybersecurity administrators.

Responsibilities in cloud services

	IaaS	PaaS	SaaS
	Sovellukset (Applications)	Sovellukset (Applications)	Sovellukset (Applications)
	Tieto (Data)	Tieto (Data)	Tieto (Data)
	Suorituspalvelu (Runtime)	Suorituspalvelu (Runtime)	Suorituspalvelu (Runtime)
	Väliohjelmisto (Middleware)	Väliohjelmisto (Middleware)	Väliohjelmisto (Middleware)
Asiakas (Customer/tenant)	Käyttöjärjestelmä (Operating system)	Käyttöjärjestelmä (Operating system)	Käyttöjärjestelmä (Operating system)
Palvelutarjoja (Provider)	Virtualisointi (Virtualisation)	Virtualisointi (Virtualisation)	Virtualisointi (Virtualisation)
	Palvelimet (Servers)	Palvelimet (Servers)	Palvelimet (Servers)
	Tallenuskapasiteetti (Storage)	Tallenuskapasiteetti (Storage)	Tallenuskapasiteetti (Storage)
	Verkko (Networking)	Verkko (Networking)	Verkko (Networking)
	Palvelinkeskus (Facility)	Palvelinkeskus (Facility)	Palvelinkeskus (Facility)

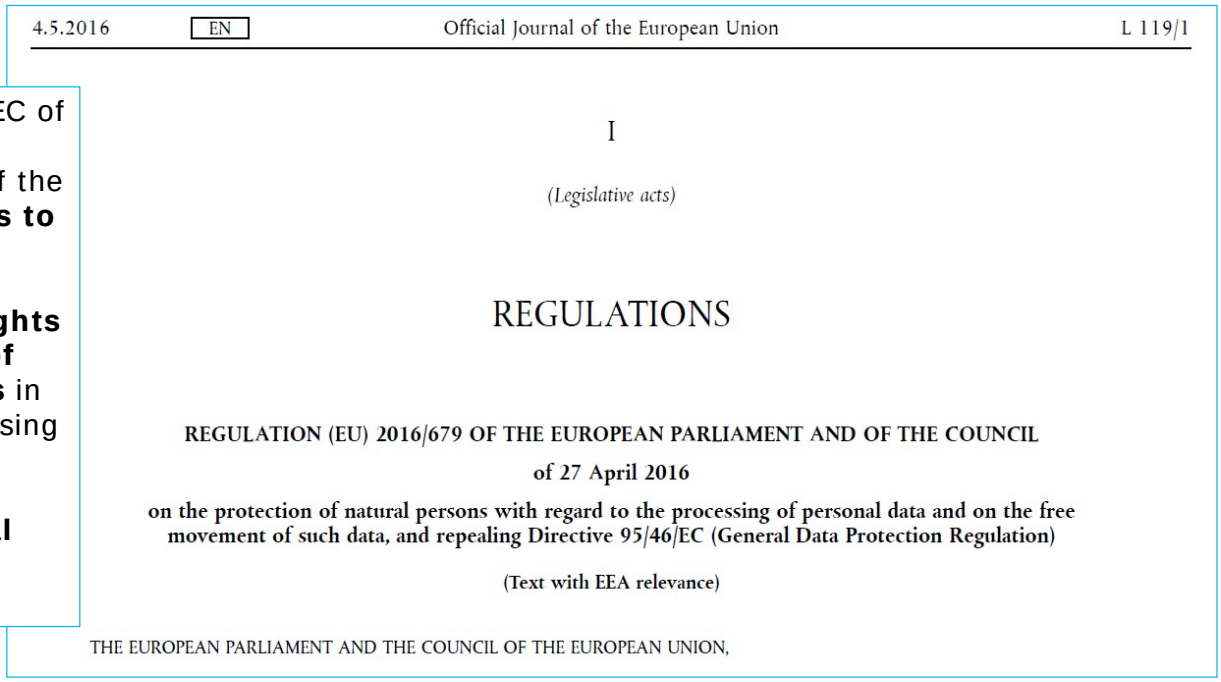
Kuva 1. Tyypillinen vastuujakomalli

[9] National Institute of Standards and Technology (NIST). 2011. Special Publication 800-145: The NIST Definition of Cloud Computing. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf.

GDPR

GDPR = General Data Protection Regulation

Directive 95/46/EC of the European Parliament and of the Council (4) **seeks to harmonise the protection of fundamental rights and freedoms of natural persons** in respect of processing activities and **to ensure the free flow of personal data** between Member States.



4.5.2016 EN Official Journal of the European Union L 119/1

I

(Legislative acts)

REGULATIONS

REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL

of 27 April 2016

on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)

(Text with EEA relevance)

THE EUROPEAN PARLIAMENT AND THE COUNCIL OF THE EUROPEAN UNION,

benchmarking

benchmarking (FI: vertailu/vertais analyysi/kehittäminen, esikuva/valio -analyysi)

3.362 , benchmark

1. standard against which results can be measured or assessed

2. procedure, problem, or test that can be used to compare systems or components to each other or to a standard

3. reference point against which comparisons can be made

3.363 , benchmarking

1. activity of comparing objects of interest to each other or against a benchmark to evaluate characteristic(s)

2. the comparison of actual or planned practices, such as processes and operations, to **those of comparable organizations to identify best practices**, generate ideas for improvement, and provide a basis for measuring performance.

Note 1 to entry: In the context of ISO/IEC 29155, the object of interest is IT project performance, and the characteristic is a particular aspect of an IT project such as productivity.

[ISO 24765:2017]

CMM

CMM / CMMI [SEI/CMU]

CMM = Capability Maturity Model.

CMMI (Capability Maturity Model Integration) models are **collections of best practices** that help organizations **to improve their processes**. These models are developed by product teams with members from industry, government, and the Software Engineering Institute (SEI).

The CMMI-DEV model provides guidance for applying CMMI best practices in a development organization. Best practices in the model focus on activities for developing quality products and services to meet the needs of customers and end users.

There are many CMM-*

The original SEI/CMU CMM levels, 1993

Organisation could do self-assesment. All process parts should be on a certain level, to get a sertification. Some processes may be at higher level.

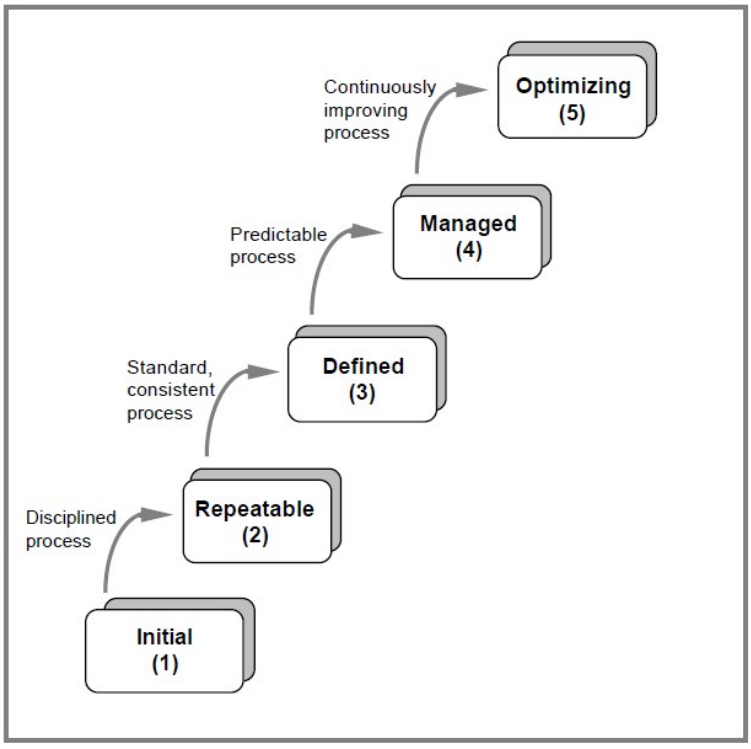
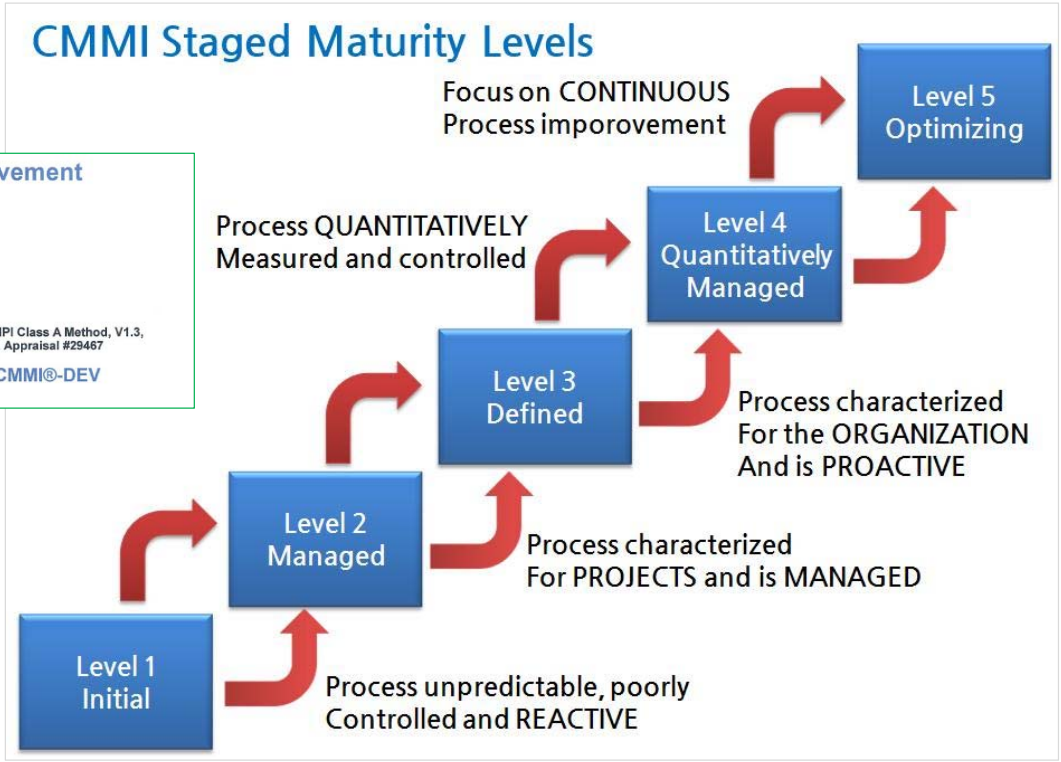
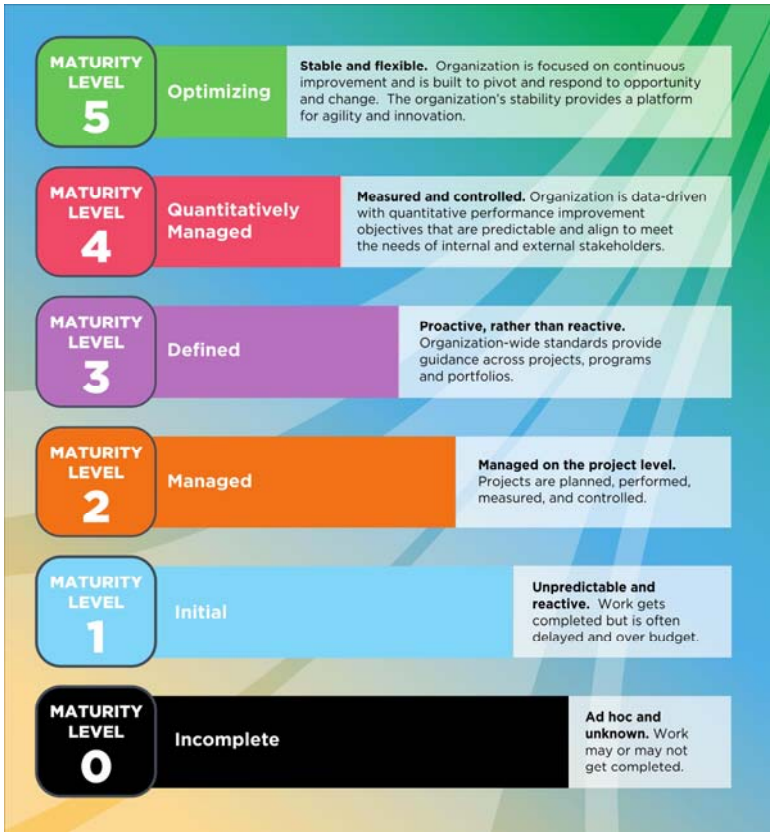


Figure 2.2 The Five Levels of Software Process Maturity



[https://iqmsglobal.com/cmmi/]

CMMI levels



[cmmiinstitute.zendesk.com]

Highlights - What to remember

- embedded software systems must keep both sw and hw development in control
- use state diagrams for help
- a lot of hype at IoT and AI at the moment; few experts available
- many Finnish and European Union projects started and ongoing about IoT and AI
- cyber security needs consideration in all software development

- few university courses about IoT and AI; at TUNI Spring 2020